Revisiting Leakage-Resilient MACs and Succinctly-Committing AEAD

More Applications of Pseudo-Random Injections

Mustafa Khairallah

Dept. of Electrical and Information Technology, Lund University, Lund, Sweden
mustafa.khairallah.1608(at)eit.lth.se

Abstract. Pseudo-Random Injections (PRIs) have had several applications in symmetric-key cryptography, such as in the idealization of Authenticated Encryption with Associated Data (AEAD) schemes, building robust AEAD, and, recently, in converting a committing AEAD scheme into a succinctly committing AEAD scheme. In Crypto 2024, Bellare and Hoang showed that if an AEAD scheme is already committing, it can be transformed into a succinctly committed scheme by encrypting part of the plaintext using a PRI. In this paper, we revisit the applications of PRIs in building Message Authentication Codes (MACs) and AEAD schemes. First, we look at some of the properties and definitions PRIs, such as collision resistance and unforgeability when used as a MAC with small plaintext space, under different leakage models. Next, we show how they can be combined with collision-resistant hash functions to build a MAC for long plaintexts, offering flexible security depending on how the PRI and equality check are implemented. If both the PRI and equality check are leak-free, the MAC provides almost optimal security, but the security only degrades a little if the equality check is only leakage-resilient (rather than leak-free). If the equality check has unbounded leakage, the security drops to a baseline security, rather than being completely insecure. Next, we show how to use PRIs to build a succinctly committing online AEAD scheme dubbed as scoAE from scratch that achieves succinct CMT4 security, privacy, and Ciphertext Integrity with Misuse and Leakage (CIML2) security. Last but not least, we show how to build a succinct nonce Misuse-Resistant (MRAE) AEAD scheme, dubbed as scMRAE. The construction combines the SIV paradigm with PRI-based encryption (*e.g.* the Encode-then-Encipher (EtE) framework).

Keywords: Context Commitment · Succinct · AEAD · MAC · Leakage Resilience

1 Introduction

Authenticated Encryption with Associated Data (AEAD) has become the defacto symmetric-key encryption notion, as it provides both confidentiality and authenticity, simultaneously. As AEAD has become widespread, new threats emerged, such as nonce repetition [RS06], leakage-based attacks [BBC+20] and attacks on context commitment [ADG+22]. Recently, the relation between leakage-resilient AEAD and context-committing AEAD has become a topic of study. Struck and Weishäupl [SW24] studied the context-commitment of generic AEAD constructions and how it relates to the construction of leakage-resilient AEAD. Later, Dhar *et al.* [DEJ+24] studied the context commitment of prominent leakage-resilient schemes, showing that several of these schemes are already context-committing with security up to half the tag size.

In Crypto 2024 [BH24], Bellare and Hoang studied an issue that arises in context-committing schemes, especially tag-based AEAD schemes. These schemes offer context commitment security only up to half their tag sizes (alternatively known as ciphertext

expansion). They defined a *succinctly committing* AEAD scheme as a context-committing AEAD scheme with security higher than half the ciphertext expansion. They proposed a transformation from a context-committing AEAD scheme to a succinctly committing AEAD scheme. If the underlying AEAD scheme is tag-based but not context-committing, then we can use another transformation to make it context-committing before applying this transformation.

In order to explain their solution and related solutions, we recall what a tag-based scheme is. From a high-level perspective, a tag-based AEAD scheme encrypts each message into a variable-length ciphertext and a fixed-length tag. During decryption, the ciphertext is used to derive a plaintext and a fixed-length tag, which is compared to the tag provided by the user, and the plaintext is released if and only if the tags match. Thus, such schemes cannot offer committing security beyond half the tag size as the adversary can simply attempt to find tag collisions. Consequently, the output of a succinctly committing AEAD scheme cannot be separated into tags and ciphertexts. What Bellare and Hoang [BH24] propose is to divide the plaintext M into two parts: M' and $M^\star$, where only the former is encrypted using a tag-based context-committing scheme, and the tag is used as a key to encrypt the latter using a Pseudo-Random Injection (PRI) (which they call an invertible pseudo-random function). While this approach improves context commitment significantly, it requires an underlying AEAD scheme that is already committing. If the scheme is not already committing, we have to apply two transformations, each with its own overhead.

One may consider using a wide block cipher in the Encode-then-Encipher (EtE) framework [HKR15]. This resolves the issue of collision-finding attacks on the tag, as there is no distinct tag, but the whole ciphertext is needed for authenticity. However, it has been shown that most practical realizations of this strategy do not offer high commitment as one would expect from an ideal wide block cipher [CFGI+23]. Recently, Naito *et al.* [NSS24] proposed an EtE-based AEAD scheme that is succinctly committing. Their solution assumes the existence of a non-committing wide block cipher, and applies a transformation inspired by that of [BH24] on top of it. While this solution does not assume an already committing scheme, it targets higher security than what we expect from a standard AEAD scheme, and uses a wide block cipher which is typically an expensive primitive compared to AEAD.

Contributions The goal of this work is to explore the applications of PRIs in building more flexible and more efficient symmetric key algorithms. First, we explore some of the properties of PRIs. We show that an ideal PRI is collision resistant with a similar bound to that given by Bellare and Hoang [BH24] for a PRI built from an ideal Tweakable Block Cipher (TBC) using EtE. We also study their strong unforgeability with leakage as MACs under different assumptions: leak-free, unpredictable with leakage and/or leakage-resilient value comparison [DM21] implementations.

Next, we study the application of PRIs in building a collision-resistant, leakage-resilient MAC that has the flexibility of providing security vs. efficient trade-offs. We observe that LRMAC1, proposed by Berti *et al.* can be interpreted as being based on a PRI where the plaintext space of the PRI is limited to only the all-zero vector. On the other hand, the collision resistance of LRMAC1 was studied in [DEJ+24]. We propose iLRMAC[1], a generalization of LRMAC1 where the TBC is replaced with a general PRI. We show that it inherits the collision resistance of the PRI and the hash function used. We also study its strong Unforgeability with decryption Leakage (sUF-L2). In LRMAC1, the inverse PRI can either return 0^n or $\perp$. In our case, the inverse PRI can return multiple values, and we need to compare these values with a value given by the user. We can reduce the unforgeability with leakage of the full MAC to that of the underlying PRI. As a side note, we demonstrate an error in the original interpretation of the bounds given in [BGPS21],

[1] the i stands for *"injective"*.

where the authors claimed to achieve beyond birthday bound security, but their main theorem did not support such claim.

Next, we propose two new AEAD schemes. The first scheme is dubbed succinctly committing online AEAD (scoAE). It is an idealization of a wide class of online AEAD schemes, where the nonce, associated data and most of the plaintext are absorbed by a keyed encryption function. This encryption function also generates an auxiliary output which is collision-resistant: for the *same key*, it is hard to find two sets of inputs where the auxiliary output is the same. The auxiliary output is used a tweak for a keyed PRI that encrypts the last m bits of the plaintext. The output of the PRI is both a *tag* and a ciphertext of the last m bits of the plaintext. This approach improves on [BH24] in two regards: this AEAD construction is both online (if the encryption function is online) and does not require a transformation, avoiding the nested assumptions used in [BH24]. Besides, as the *hashed* auxiliary output of the encryption function is only used as a tweak, and not as a key, we can instantiate the scheme without any randomness assumptions on the auxiliary output. We show that such construction is CIML2-secure under reasonable leakage assumptions: heavily-protected PRI, and the encryption function is split into a heavily-protected key derivation function and a collision-resistant function with unbounded leakage. This shows that a duplex sponge construction or an encrypt-then-MAC construction can be easily converted into being succinctly committing with minimal modifications.

The second AEAD scheme we propose is dubbed succinctly committing Misuse-Resistant AEAD (scMRAE). It is a nonce-misuse-resistant scheme based on similar techniques but with a two-pass MAC-then-Encrypt structure, where the MAC is similar to iLRMAC, with one less check, and can achieve Misuse-Resistant AEAD (MRAE) security and CMT4 security. The idea is mix both that EtE approach and the Synthetic (SIV) approach. We divide the plaintext M into $(M', M^\star)$ where $M^\star$ is the last m-bit string of M. By using an iLRMAC-like structure for the MAC layer, the tag becomes an EtE-like encryption of $M^\star$. Then, the tag is used as an Initial Vector (IV) for a stream cipher that encrypts M' into C'. During decryption, first, we recover M' from C' and the tag. Then, we verify that for any $M^\star$, the tag is valid; this process also returns the valid $M^\star$, if any. If there is no such $M^\star$, the ciphertext is deemed invalid. It is easy to follow that this scheme works closely to SIV, where any change in any of the inputs N, A or M affects the tag and the ciphertext, ensuring privacy with nonce-misuse. The integrity (with and out leakage) is similar to the unforgeability of iLRMAC without value comparison: we just require that the tag corresponds to any valid PRI point. Similarly, the context commitment reduces to the collision resistance of iLRMAC.

We emphasize that succinct commitment is different from the concept of compact commitment introduced in [GLR17], which refers to a scheme where only a constant part of the ciphertext needs to be checked to verify commitment. A scheme can be succinctly committing but not compactly committing, and vice versa. Our schemes, as well as the transformation of [BH24], are both succinctly committing and compactly committing.

2 Preliminaries

General Notation For a set $\mathcal{X}$, we write $X \xleftarrow{\$} \mathcal{X}$ to denote that a value X is sampled uniformly at random from $\mathcal{X}$. $\{0,1\}^*$ is the set of all bit strings, including the empty string ϵ. $\{0,1\}^b$ is the set of bit strings of length b and $\{0,1\}^{\leq b}$ is the set of bit strings of length at most b, including the empty string. For an integer $m > 0$, we say $(M^\star, M') \leftarrow \mathtt{parse}(M)$, such that $M^\star = M$ and $M' = \epsilon$ if $M \in \{0,1\}^{\leq m}$, and $M'\|M^\star = M$ such that $|M^\star| = n$, otherwise. An adversary $\mathcal{A}$ is a computationally bounded algorithm that plays a security game against a challenger. We indicate that $\mathcal{A}$ outputs X by $X \leftarrow \mathcal{A}$. We say $|X|$ to represent the bit length of a the bit string X. We say $|\mathcal{X}|$ to represent the number of elements in the set $\mathcal{X}$. If x' is a linear function of x, we say $x' = \mathsf{lin}(x)$.

Collision Resistance Let $H : \mathcal{K}_h \times \mathcal{M} \to \mathcal{X}$ be a hash function. A hash function H is called $(\varepsilon_{\mathsf{cr}}, t)$-collision-resistant if for every t-bounded adversary $\mathcal{A}$ (*i.e.* running in time at most t), the probability that $\mathcal{A}(s)$ outputs a pair of distinct inputs $(M_1, M_2) \in \mathcal{M}^2$, such that $H_s(M_1) = H_s(M_2)$ and $M_1 \neq M_2$, is bounded by ε_{cr}, with $s \xleftarrow{\$} \mathcal{K}_h$ picked uniformly at random:

$$Pr[s \xleftarrow{\$} \mathcal{K}_h, (M_1, M_2) \leftarrow \mathcal{A}(s) \in \mathcal{M}^2 \; s.t. \; M_1 \neq M_2, H_s(M_1) = H_s(M_2)] \leq \varepsilon_{cr}.$$

The following notions and constructions use a collision-resistant hash function as a building block. Its key is shared with the adversary at the the beginning of the game, and is included in the syntax definition of respective constructions. This explains why some definitions include two key domains: $\mathcal{K}_h$ as the domain of the hash key, and $\mathcal{K}$ as the domain of the secret key. In collision games, the secret key is treated as part of the chosen input.

Collision-Resistant Encryption Let $E : \mathcal{K}_h \times \mathcal{K} \times \mathcal{N} \times \mathcal{A} \times \mathcal{M} \to \mathcal{C} \times \mathcal{V}$ be an encryption function with auxiliary output. An encryption function with auxiliary output E is called $(\varepsilon_{\mathsf{cr}}, t)$-collision-resistant if for every t-bounded adversary $\mathcal{A}$ (*i.e.* running in time at most t), the probability that $\mathcal{A}(s)$ outputs a key K and a pair of distinct inputs $(M_1, M_2) \in \mathcal{M}^2$, such that $V_1 = V_2$, $(C_1, V_1) \leftarrow E_s(K, M_1)$ and $(C_1, V_1) \leftarrow E_s(K, M_2)$, is bounded by ϵ_{cr}, with $s \xleftarrow{\$} \mathcal{K}_h$ picked uniformly at random:

$$Pr[s \xleftarrow{\$} \mathcal{K}_h, (K, N_1, A_1, M_1, N_2, A_2, M_2) \leftarrow \mathcal{A}(s) \; s.t. \; (N_1, A_1, M_1) \neq (N_2, A_2, M_2),$$

$$E_s(K, N_1, A_1, M_1) = (C_1, V_1), E_s(K, N_2, A_2, M_2) = (C_2, V_2), V_1 = V_2] \leq \varepsilon_{cr}.$$

We drop the suffix s when clear from the context. We shall also call an encryption function with auxiliary output E is called $(\varepsilon_{\mathsf{cr}}, t)$-strongly-collision-resistant if for any t-bounded adversary $\mathcal{A}$

$$Pr[s \xleftarrow{\$} \mathcal{K}_h, (K_1, N_1, A_1, M_1, K_2, N_2, A_2, M_2) \leftarrow \mathcal{A}(s)$$

$$s.t. \; (K_1, N_1, A_1, M_1) \neq (K_2, N_2, A_2, M_2),$$

$$E_s(K_1, N_1, A_1, M_1) = (C_1, V_1), E_s(K_2, N_2, A_2, M_2) = (C_2, V_2), V_1 = V_2] \leq \varepsilon_{cr}.$$

Message Authentication Codes (MACs) Let $\mathsf{Mac} : \mathcal{K}_h \times \mathcal{K} \times \mathcal{M} \to \mathcal{T}$ be a function that takes as input a hash key $s \in \mathcal{K}_h$, a secret key $K \in \mathcal{K}$ and message $M \in \mathcal{M}$ and returns a tag $T \in \mathcal{T}$. $\mathsf{Ver} : \mathcal{K}_h \times \mathcal{K} \times \mathcal{M} \times \mathcal{T} \to \{\mathtt{true}, \mathtt{false}\}$ takes the keys, message and a tag $T \in \mathcal{T}$, and returns either `true` or `false`

Correctness $\mathsf{Ver}(s, K, M, T)$ returns `true` if and only if $\mathsf{Mac}(s, K, M) = T$. We drop the hash key when it is clear from context.

strong Unforgeability with verification Leakage (sUF-L2) We follow the formalization of Berti *et al.* [BGPS21]. Let L_M be the leakage function corresponding to running Mac with secret key K and L_V be the leakage function corresponding to running Ver with secret key K. We say that Mac is $(\varepsilon, q_{\mathsf{L}}, q_m, q_v, t)$-$\mathsf{sUF\text{-}L2}$-secure against adaptive adversaries if for all adversaries that are bounded by time t and make q_{L} profiling queries to either L_M or L_V with chosen key, q_m queries to Mac and q_v queries to Ver, and does not make trivial queries:

$$\Pr[s \xleftarrow{\$} \mathcal{K}_h, K \xleftarrow{\$} \mathcal{K} : (M, T) \leftarrow \mathcal{A}^{L_M, L_V, \mathsf{Mac}, \mathsf{Ver}}(s) | \mathsf{Ver}(M, T) = \mathtt{true}] \leq \varepsilon.$$

Collision Resistance We say Mac is $(\varepsilon_{\mathsf{cr}}, t)$-collision-resistant if for every t-bounded adversary $\mathcal{A}$, the probability that $\mathcal{A}(s)$ outputs a pair of distinct inputs $((K_1, M_1), (K_2, M_2)) \in (\mathcal{K} \times \mathcal{M})^2$, such that $\mathsf{Mac}(K_1, M_1) = \mathsf{Mac}(K_2, M_2)$ and $(K_1, M_1) \neq (K_2, M_2)$, is bounded by ε_{cr}, with $s \xleftarrow{\$} \mathcal{K}_h$ picked uniformly at random:

$$Pr[s \xleftarrow{\$} \mathcal{K}_h, ((K_1, M_1), (K_2, M_2)) \leftarrow \mathcal{A}(s) \in (\mathcal{K} \times \mathcal{M})^2$$

$$s.t.\ (K_1, M_1) \neq (K_2, M_2), \mathsf{Mac}(K_1, M_1) = \mathsf{Mac}(K_2, M_2)] \leq \varepsilon_{cr}.$$

Authenticated Encryption with Associated Data (AEAD) An AEAD scheme is a pair of functions $\Pi = (\mathsf{E}, \mathsf{D})$. $\mathsf{E} : \mathcal{K} \times \mathcal{N} \times \mathcal{D} \times \{0,1\}^* \rightarrow \{0,1\}^*$ is the encryption function that takes secret key $K \in \mathcal{K}$, nonce $N \in \mathcal{N}$, associated data $A \in \mathcal{D}$ and plaintext $M \in \{0,1\}^*$ and returns ciphertext $C \in \{0,1\}^*$. $\mathsf{D} : \mathcal{K} \times \mathcal{N} \times \mathcal{D} \times \{0,1\}^* \rightarrow \{0,1\}^* \cup \{\perp\}$ is the decryption function that takes secret key $K \in \mathcal{K}$, nonce $N \in \mathcal{N}$, associated data $A \in \mathcal{D}$ and ciphertext $C \in \{0,1\}^*$ and returns plaintext $M \in \{0,1\}^*$ or the symbol $\perp$.

Ciphertext Expansion Let $\mathsf{E}(K, N, A, M) = C$, then $l = |C| - |M| > 0$ is known as the ciphertext expansion.

Correctness and Tidiness If the scheme satisfies that

$$\mathsf{D}(K, N, A, \mathsf{E}(K, N, A, M)) = M$$

for all inputs, we say the scheme is correct. If the scheme satisfies that

$$\mathsf{E}(K, N, A, \mathsf{D}(K, N, A, C)) = C$$

for all inputs such that $\mathsf{D}(K, N, A, C) \neq \perp$, we say the scheme is tidy.

Confidentiality Let $\mathcal{A}$ be an adversary that makes q queries to E with secret key K, then outputs either 0 or 1. We say the scheme ciphertexts are indistinguishable from random strings against chosen plaintext adversaries. We define the advantage of $\mathcal{A}$ as

$$\mathrm{Adv}^{\mathsf{indcpa}}_{\Pi}(\mathcal{A}) \stackrel{\text{def}}{=} |\Pr[K \xleftarrow{\$} \mathcal{K} : 1 \leftarrow \mathcal{A}^{\mathsf{E}}] - \Pr[1 \leftarrow \mathcal{A}^{\$}]|,$$

where \$ returns a uniformly random string of the correct ciphertext length for every query, assuming $\mathcal{A}$ does not repeat queries. We refer to [BBC+20] for a detailed discussions of generalizations of this security notion to include different types of leakage and nonce handling. We note that if the scheme is secure against nonce repeating adversaries, we say the scheme achieves misuse-resistant confidentiality. We will only deal with black-box confidentiality, *i.e.*, without any leakage.

Ciphertext Integrity with nonce Misuse and Leakage (CIML2) Let $\mathcal{B}$ be an adversary that makes q_e queries to E with an associated leakage function L_e and q_d queries to D with an associated leakage function L_d. Let $\mathcal{B}$ does not make trivial queries but can repeat nonces in both encryption and decryption queries. We say Π is $\mathsf{CIML2}$-secure if the probability that $\mathcal{B}$ forges Π, *i.e.*, any decryption query returns a value other than $\perp$, is negligible:

$$\mathrm{Adv}^{\mathsf{ciml2}}_{\Pi}(\mathcal{B}) \stackrel{\text{def}}{=} |\Pr[K \xleftarrow{\$} \mathcal{K} : 1 \leftarrow \mathcal{B}^{\mathsf{E},\mathsf{D},L_e,L_d}\ forges\ \Pi]|$$

If a scheme achieves both confidentiality and integrity against nonce-repeating adversaries, we say the scheme is MRAE secure.

CMT4 Security Bellare and Hoang [BH22] studied the relations between different context commitment security notions and showed that CMT4 is the strongest notions for correct and tidy schemes, and we shall focus on it. In the CMT4 game against an AE scheme Π, an adversary $\mathcal{C}$ outputs (K_1, N_1, A_1, M_1) and (K_2, N_2, A_2, M_2); $\mathcal{C}$ wins if:

- $(K_1, N_1, A_1, M_1) \neq (K_2, N_2, A_2, M_2)$;
- $\mathsf{E}(K_1, N_1, A_1, M_1) = \mathsf{E}(K_2, N_2, A_2, M_2)$.

We write $\varepsilon_{\mathsf{CMT4}}$ to denote the upper bound on the probability that any such adversary wins. The adversary has access to the ideal primitives and hash keys used by Π.

3 Pseudo-Random Injections

As cryptographic primitives, PRIs are not as widely used as other cryptographic primitives such PRFs or PRPs. However, they have seen applications in idealizing AEAD [RS06, Kha24] and the design of robust AEAD schemes [FLPQ13, BF18]. In Crypto 2024, Bellare and Hoang [BH24] used a restricted version of a PRI as part of a construction to transform a tag-based committing AEAD scheme into a succinctly committing AEAD scheme. In the rest of this paper, we shall use PRIs to construct a collision-resistant MAC and an adhoc succinctly committing AEAD, *i.e.*, succinctly committing AEAD schemes that are not based on an underlying tag-based AEAD scheme. In this section, we establish some of the definitions and properties needed.

Definition 1. A keyed tweakable injective function $f : \{0,1\}^k \times \{0,1\}^h \times \mathcal{M} \to \mathcal{C}$ is a function with key space $\{0,1\}^k$, tweak space $\{0,1\}^h$, plaintext space $\mathcal{M}$ and ciphertext space $\mathcal{C}$, such that $|\mathcal{M}| \leq |\mathcal{C}|$. For any $(K, H) \in \{0,1\}^k \times \{0,1\}^h$, $f(K, H, \cdot)$ is an injective function from $\mathcal{M}$ to $\mathcal{C}$. $f^{-1} : \{0,1\}^k \times \{0,1\}^h \times \mathcal{C} \to \{\perp\} \cup \mathcal{M}$ is its inverse, such that $f^{-1}(K, H, C) = M$ if and only if $f(K, H, M) = C$. $f^{-1}(K, H, C) = \perp$ if $\forall M \in \mathcal{M}$, $f(K, H, M) \neq C$. If $\mathcal{M} = \mathcal{C}$, then f is a Tweakable Block Cipher (TBC).

Definition 2. An function $f : \{0,1\}^k \times \{0,1\}^h \times \mathcal{M} \to \mathcal{C}$ is called a $(\varepsilon_{\mathsf{pri}}, t)$-secure PRI if, for any distinguishing adversary $\mathcal{A}$ that runs in time at most t,

$$|\Pr[\tilde{\pi} \xleftarrow{\$} \mathcal{F}_{h,\mathcal{M},\mathcal{C}} : 1 \leftarrow \mathcal{A}^{\tilde{\pi},\tilde{\pi}^{-1}}] - \Pr[K \xleftarrow{\$} \{0,1\}^k : 1 \leftarrow \mathcal{A}^{f(K,\cdot,\cdot),f^{-1}(K,\cdot,\cdot)}]| \leq \varepsilon_{\mathsf{pri}},$$

where $\mathcal{F}_{h,\mathcal{M},\mathcal{C}}$ is the set of all injections from $\{0,1\}^h \times \mathcal{M}$ to $\mathcal{C}$.

If the PRI is replaced by a TBC, we shall refer to the security bound as $\varepsilon_{\mathsf{stprp}}$.

Definition 3. An ideal PRI $f : \{0,1\}^k \times \{0,1\}^h \times \mathcal{M} \to \mathcal{C}$ is a family of keyed tweakable injections indexed by $K \in \{0,1\}^k$, sampled uniformly randomly from the set of all possible families of keyed tweakable injections with the same parameters. In other words, $\forall K \in \{0,1\}^k$, $f(K, \cdot, \cdot)$ is sampled uniformly at random from $\mathcal{F}_{h,\mathcal{M},\mathcal{C}}$. If $\mathcal{M} = \mathcal{C}$, then f is an ideal TBC. Such PRI constructed using lazy sampling is depicted in Algorithm 1, where $\mathcal{M} = \{0,1\}^{\leq m}$ and $\mathcal{C} = \{0,1\}^n$.

3.1 Collision Resistance

Proposition 1. *Let $f : \{0,1\}^k \times \{0,1\}^h \times \{0,1\}^{\leq m} \to \{0,1\}^n$ be an ideal PRI. Then, f is $(\varepsilon_{\mathsf{cr}}, t)$-collision-resistant such that*

$$\varepsilon_{\mathsf{cr}} \leq \frac{q_e^2 + 2}{2^n} + \frac{4q_d}{2^{n-m}}$$

where any adversary makes at most q_e queries to f and q_d queries to f^{-1}, and $t = O(t_{q_e} + t_{q_d})$, t_{q_e} is the time needed to make q_e queries to f and t_{q_d} is the time needed to make q_d queries to f^{-1}. The adversary makes at most $(q_d + q_d) < 2^{n-1}$ queries.

Algorithm 1 An ideal PRI implemented using lazy sampling.

for $(K, H) \in \{0,1\}^k \times \{0,1\}^h$ **do**
  $\mathsf{Dom}(K, H) \leftarrow \phi$
  $\mathsf{Img}(K, H) \leftarrow \phi$
  $\mathsf{Invalid}(K, H) \leftarrow \phi$
end for
$f(K, H, M)$:
$C \xleftarrow{\$} \{0,1\}^n \setminus (\mathsf{Img}(K, H) \cup \mathsf{Invalid}(K, H))$
$\mathsf{Img}(K, H) \leftarrow \mathsf{Img}(K, H) \cup \{C\}$
$\mathsf{Dom}(K, H) \leftarrow \mathsf{Dom}(K, H) \cup \{M\}$
return C
$f^{-1}(K, H, C)$:
$\mathsf{EligY} \leftarrow \{0,1\}^n \setminus (\mathsf{Img}(H, K) \cup \mathsf{Invalid}(K, H))$
$\mathsf{EligX} \leftarrow \{0,1\}^{\leq m} \setminus \mathsf{Dom}(H, K)$
$x \xleftarrow{\$} [1 \ldots |\mathsf{EligY}|]$
if $x > |\mathsf{EligX}|$ **then**
  $\mathsf{Invalid}(K, H) \leftarrow \mathsf{Invalid}(K, H) \cup \{C\}$
  return $\perp$
else
  $X \xleftarrow{\$} \{0,1\}^{\leq m} \setminus \mathsf{Dom}(K, H)$
  $\mathsf{Img}(K, H) \leftarrow \mathsf{Img}(K, H) \cup \{C\}$
  $\mathsf{Dom}(K, H) \leftarrow \mathsf{Dom}(K, H) \cup \{X\}$
  return X
end if

Proof. An ideal PRI implemented using lazy sampling is given in Algorithm 1. Let $\mathcal{A}$ be an adversary that runs in time at most t, makes q_e queries to f and q_d queries to f^{-1} and returns (K_1, H_1, M_1) and (K_2, H_2, M_2). If $(K_1, H_1) = (K_2, H_2)$, then $f(K_1, H_1, M_1) = f(K_1, H_1, M_2)$ if and only if $M_1 = M_2$, which is not a valid challenge. Thus, for a valid challenge, $(K_1, H_1) \neq (K_2, H_2)$ must hold. Next, we describe a sequence of hybrid games, where E_i is the event that the adversary wins in game i. Game 0 is the game where the oracles are described according to Algorithm 1. In game 1, an adversary $\mathcal{B}$ accepts queries from $\mathcal{A}$, queries f/f^{-1} and passes the response back to $\mathcal{A}$. $\mathcal{B}$ also keeps a query table, and terminates the game if

- Two queries to f lead to the same response. For all the pairs where $(K_1, H_1) = (K_2, H_2)$, this event is impossible. Thus, the probability of this event is maximized when (K, H) is unique for all queries, which is bounded by $\binom{q_e}{2}/2^n$.
- A query to f^{-1} returns $M \neq \perp$.

For the second event, assume that the first $i-1$ queries to f^{-1} all returned $\perp$. The i^{th} query is (K_i, H_i, C_i). The probability of $M_i \neq \perp$ is

$$\leq \frac{|\mathsf{EligX}|}{|\mathsf{EligY}|}.$$

We know that $|\mathsf{EligX}| \leq 2^{m+1} - 1$. On the other hand, $|\mathsf{EligY}| = 2^n - q_e^{K_i,H_i} - q_d^{K_i,H_i}$, where $q_e^{K_i,H_i}$ and $q_d^{K_i,H_i}$ are the number of of queries that share the same (K_i, H_i) as the i^{th} query. Let $q_e^{K_i,H_i} + q_d^{K_i,H_i} \leq 2^{n-1}$, then

$$\frac{|\mathsf{EligX}|}{|\mathsf{EligY}|} \leq \frac{2(2^{m+1} - 1)}{2^n} \leq \frac{4}{2^{n-m}}.$$

Thus, the second event is bounded by a simple hybrid argument by $4q_d/2^{n-m}$, and

$$|\Pr[E_0] - \Pr[E_1]| \leq \frac{q_m^2}{2^n} + \frac{4q_d}{2^{n-m}}.$$

If $\mathcal{B}$ does not terminate, then the challenge can succeed only if at least one of (K_1, H_1, M_1) and (K_2, H_2, M_2) has not appeared in any previous query and $(K_1, H_1) \neq (K_2, H_2)$. Since $f(K_1, H_1, M_1)$ and $f(K_2, H_2, M_2)$ are the outputs of two independent and uniformly random permutations, without loss of generality, we assume that (K_2, H_2, M_2) did not appear in any previous query, and

$$\Pr[E_1] \leq \frac{1}{2^n - q_e^{K_2,H_2} - q_d^{K_2,H_2}} \leq \frac{2}{2^n}.$$

Finally,

$$\varepsilon_{\mathsf{cr}} \leq \Pr[E_1] + |\Pr[E_1] - \Pr[E_0]| \leq \frac{q_e^2 + 2}{2^n} + \frac{4q_d}{2^{n-m}}.$$

□

Comments and comparison to [BH24] The bound in Proposition 1 is similar to the bound given by Bellare and Hoang [BH24, Proposition 5.4] but for any ideal PRI rather than EtE. In [BH24], Bellare and Hoang gave an invertible PRF construction that they dubbed Hash-then-Mask (HtM). The construction is essentially a PRI when K is secret and random, but it is not an ideal PRI when the key can be chosen. However, they provided a security proof for its collision resistance. The construction can be seen as $f : \{0,1\}^k \times \{0,1\}^{\leq m} \to \{0,1\}^{b+\tau}$, where b and k are the block size and the key size of an underlying block cipher, respectively, and $m, \tau < b$ are parameters of the scheme. They showed that for the HtM construction,

$$\varepsilon_{\mathsf{cr}} \leq \frac{4(b+\tau)(q_e + q_d) + 5}{2^\tau}$$

which limits the security to $\tau - log(b+\tau)$ bits. The optimal security, according to Proposition 1, is $\min((b+\tau)/2, b+\tau-m)$. Since $\tau < b$, then $(b+\tau)/2 > \tau$. If $m \approx \tau$, then $b+\tau-m \approx b$. In all the cases, we can show that as long as $b-m$ and $b-\tau$ are less than $log(b+\tau)$, then HtM is close to optimal, with gap at most $2log(b+\tau)$.

Another observation is that, even in the ideal case, we have a term on the form $q/2^{n-m}$, which can be problematic if m is large. Even if m is relatively small, the advantage is larger than an optimal compression function. When we refer to Proposition 1 as the optimal bound, we are restricting ourselves to ideal functions according to Definition 3. The reason for that is two-fold:

1. The construction is expected to produce the output of a PRF or an AEAD scheme. Thus, the output should be indistinguishable from random when the key is secret and uniform. Optimal, in this scenario, refers to a function that is indistinguishable from random for every key selection.

2. One may envision pathological constructions that do not allow collisions, or have better bounds. Not only are these constructions not likely to be indistinguishable from random, but also they are hard to define when $h+k$ is much larger than n.

3.2 strong Unforgaebility with verification Leakage (sUF-L2):

This security notion was introduced by Berti *et al.* [BGPS21] for MACs. The adversary interacts with a MAC function using two oracles: Mac to generate tags, and Ver to verify tags. Each oracle has an associated leakage function L_m and L_v, respectively. The adversary also has access to an *offline* leakage function L that it can query with chosen key. The adversary does not make trivial queries and wins if any of the queries to Ver returns `true`. Now we define a MAC based on the PRI as follows:

Definition 4. Let the plaintext space be $\{0,1\}^h \times \mathcal{M}$ and the tag space is $\mathcal{C}$. $\mathsf{Mac}(K, H, M)$ returns $(f(K, H, M), \mathsf{L}_e(K, H, M))$. $\mathsf{Ver}(K, H, M, C)$ returns $(\mathsf{true}, \mathsf{L}_d(K, H, M, C))$, if

$$f^{-1}(K, H, C) = M,$$

and $(\mathsf{false}, \mathsf{L}_d(K, H, M, C))$, otherwise.

We say an implementation of the MAC in Definition 4 is $(\varepsilon_{\mathsf{sUF-L2}}, q_{\mathsf{L}}, q_e, q_d, t)$-sUF-L2 secure if for any adversary $\mathcal{A}$ that makes q_{L} offline queries to L, q_e queries to Mac, q_d queries to Ver, and runs in time at most t,

$$\Pr[\mathcal{A}\ wins] \leq \varepsilon_{\mathsf{sUF-L2}}.$$

Depending on how the PRI and equality check are implemented and the different leakage functions, the game captures one of two adversarial:

1. It should be hard for the adversary to return a valid point (H, M, C) such that $f(K, H, M) = C$.
2. It should be hard for the adversary to return a valid tweak-ciphertext pair (H, C) such that $f^{-1}(K, H, C) \neq \perp$.

For instance, if the the implementation uses an equality check with unbounded leakage, or the syntax is restricted by not letting the adversary input a candidate plaintext, then predicting any valid unseen ciphertext of the PRI is akin to successful forgery. This can be useful when the construction is used in an AEAD scheme and the plaintext is part of the output. In the remainder of this section, we will show two strategies to achieve security against each of this goals, starting with the latter for its simplicity. Then, we will provide comments on how these PRIs may be instantiated in practice.

3.2.1 Unbounded Leakage Value Comparison

In this model, we define the leakage functions as

$$\mathsf{L}_e(K, H, M) = (H, M, f(K, H, M), \mathsf{L}_f)$$

and

$$\mathsf{L}_d(K, H, M, C) = (H, M, C, f^{-1}(K, H, C), \mathsf{L}_f).$$

In this case, the adversary can trivially win the game if it makes any query where $f^{-1}(K, H, C) \neq \perp$. L_f is the leakage associated with running either f or f^{-1} on a specified point. If $\mathsf{L}_f = \perp$ for all queries, this is defined as the leak-free model. A simple hybrid argument shows that

$$\varepsilon_{\mathsf{sUF-L2}} \leq \varepsilon_{\mathsf{pri}} + \frac{4q_d}{2^{n-m}},$$

where $q_e + q_d \leq 2^{n-1}$. However, if L_f leaks non-trivial information, the analysis depends on the implementation. In order to differentiate between the case of unbounded leakage value comparison vs. leakage-resilient value comparison, we need to define a new notion for the security of the PRI with leakage, which is inspired by the sUP-L2 model introduced in [BGPS21].

Definition 5. Let $f : \{0,1\}^k \times \{0,1\}^h \times \mathcal{M} \to \mathcal{C}$ be a PRI according to Definition 1 with associated leakage functions $L_f, L_{f^{-1}}$. We say that f is $(\varepsilon_{\mathsf{vld-L2}}, q_L, q_e, q_d, t)$-Valid-Unpredictable ($(\varepsilon_{\mathsf{vld-L2}}, q_L, q_e, q_d, t)$-vld-L2 secure) if for any adversary that makes q_{L} offline leakage queries, q_e forward queries and q_d backward queries, and runs in time at most t, and returns a pair $(H, C) \in \{0,1\}^h \times \mathcal{C}$ that did not appear in any of the queries,

$$\Pr[f^{-1}(K, H, C) \neq \perp] \leq \epsilon_{\mathsf{vld-L2}}.$$

The $\varepsilon_{\mathsf{sUF-L2}}$ bound also can be derived in a similar fashion to the proof of [BGPS21, Theorem 1].

Proposition 2. *Let f be a $(\varepsilon_{\mathsf{vld-L2}}, q_{\mathsf{L}}, q_e, q_d, t')$-Valid-Unpredictable PRI defined according to Definition 1. Then, we can build a MAC* $\mathsf{Mac} : \{0,1\}^k \times (\{0,1\}^h \times \mathcal{M}) \to \mathcal{C}$ *such that* Mac *is* $(\varepsilon_{sUF-L2}, q_{\mathsf{L}}, q_e, q_d, t)$*-sUF-L2 secure, with*

$$\varepsilon_{sUF-L2} \leq (q_d + 1)\varepsilon_{\mathsf{vld-L2}},$$

where $t' = \mathsf{lin}(t)$.

Proof. We construct the MAC in Definition 4. Let $\mathcal{A}$ be a $(q_{\mathsf{L}}, q_e, q_d, t)$-sUF-L2 adversary. Let $\mathcal{B}$ be a $(q_{\mathsf{L}}, q_e, q_d, t')$-vld-L2 adversary against f. $\mathcal{B}$ interacts with $\mathcal{A}$ and uses f to simulate the MAC. Using a hybrid argument, we consider that $\mathcal{B}$ terminates the game if during any Ver query, $\mathcal{B}$ receives a response from f^{-1} other than $\perp$. The probability that $\mathcal{B}$ terminates at query $i \leq q_d$ is bounded by $\varepsilon_{\mathsf{vld-L2}}$. Thus,

$$\Pr[\mathcal{B} \text{ terminates}] \leq q_d \varepsilon_{\mathsf{vld-L2}}.$$

If $\mathcal{B}$ does not terminate, it receives the outcome of $\mathcal{A}$; (H, C) and queries $f^{-1}(K, H, C)$. It wins if $f^{-1}(K, H, C) \neq \perp$. Thus, the overall bound is given by

$$\varepsilon_{\mathsf{sUF-L2}} \leq (q_d + 1)\varepsilon_{\mathsf{vld-L2}}.$$

□

Interpretation and Issue in [BGPS21]. If the PRI is heavily protected, then it is expected that $\varepsilon_{\mathsf{vld-L2}} \leq 4/2^{n-m} + \varepsilon_{\mathsf{pri}}$. Interestingly, the security bound in the leakage-resilient model is significantly worse that the leak-free mode. This is an issue in the security modeling rather that a drop in security. To explain this, we show an error in the interpretation provided by Berti *et al.* [BGPS21]. The authors of [BGPS21] prove that the strong unforgeability with leakage (sUF-L2) of the PRI part of their construction (LRMAC1) is bounded by

$$\varepsilon_{\mathsf{sUF-L2}} \leq (q_d + 1)\varepsilon_{\mathsf{sUP-L2}},$$

where $\varepsilon_{\mathsf{sUP-L2}}$ is the probability that an adversary interacting with a leaking TBC can predict a plaintext-ciphertext pair of the TBC that has not been queried using either the encryption of the decryption oracle, regardless of the observed queries. The authors then conclude that this bound is beyond birthday bound and in the black-box model, it implies

$$(q_d + 1)\varepsilon_{\mathsf{sUP-L2}} \leq \varepsilon_{\mathsf{stprp}} + \frac{(q_d + 1)}{2^n - q_e - q_d}.$$

Unfortunately, this conclusion is not true[2]. First, note that the above inequality has to computational security terms; $\varepsilon_{\mathsf{sUP-L2}}$ and $\varepsilon_{\mathsf{stprp}}$. On the left hand side, the computational term is multiplied by by the number of queries, while the term in the right hand side is not. This leads to several contradictions. For starters, the inequality implies that

$$\varepsilon_{\mathsf{sUP-L2}} = O(\frac{1}{2^n} + \frac{\varepsilon_{\mathsf{stprp}}}{q_d + 1}).$$

Clearly, this cannot be true as it implies that unpredictability can be improved, relative to pseudo-randomness, by increasing the number of verification queries. Moreover, we can show that the inequality does not hold for a wide selection of parameters. We shall use the case where the PRI or MAC are implemented using an ideal TBC. We consider the case where the adversary makes q_p chosen-key queries to the ideal TBC, q_e forward

[2]It is not clear to us how obvious this observation is to researchers in general, so we err on the side of caution and take some space to discuss this issue. We have disclosed this issue in October 2022 to Francesco Berti and Chun Guo, designers of LRMAC1.

construction queries and q_d backward construction queries. The ideal TBC has key size of k. It is easy to see that

$$\varepsilon_{\mathsf{sUP-L2}} \geq q_p/2^k$$

which is the lower bound from any key guessing adversary, *i.e.* brute-force attacks. Thus,

$$(q_d+1)\varepsilon_{\mathsf{sUP-L2}} \geq (q_d+1)q_p/2^k.$$

On the other hand, $\varepsilon_{\mathsf{stprp}}$ can be much smaller, and in fact for uniform adversaries it is typically assumed that when the TBC is ideal,

$$\varepsilon_{\mathsf{stprp}} \leq q_p/2^k$$

nullifying the inequality.

3.2.2 Leakage-Resilient Value Comparison with PRI

One may not be satisfied with a MAC that outputs an n-bit tag but only provides $(n-m)$-bit unforgeability. We can improve this at the implementation-level by using a better value comparison function and more secure leakage functions.

We define a leakage-resilient value comparison function following [DM21] as follows:

Definition 6. Let $\mathsf{VC} : (\mathcal{M} \cup \{\perp\}) \times \mathcal{M} \rightarrow \{\texttt{true}, \texttt{false}\}$, with associated leakage function L_{VC}, be a value comparison function. $\mathsf{VC}(X, Y) = \texttt{true}$ if $X = Y$, and `false`, otherwise.

Using Definition 6, we can redefine new leakage functions for the MAC as follows:

Definition 7. Let Mac be a MAC function defined according to Definition 4, but the equality check is replace with VC given in Definition 6. The leakage functions associated with the oracles are given by

$$\mathsf{L}_e(K, H, M) = (H, M, f(K, H, M), \mathsf{L}_f)$$

and

$$\mathsf{L}_d(K, H, M, C) = (H, M, C, \mathsf{L}_{\mathsf{VC}}, \mathsf{L}_f).$$

Note that instead of leaking the actual value of the output of f^{-1}, the oracle only leaks the associated leakage of VC. In other words, the adversary does not trivially know whether a query to f^{-1} returned a valid plaintext or not. Giving a concrete bound for $\varepsilon_{\mathsf{sUF-L2}}$ requires delicate analysis of the implementation and how the two primitives interact. However, we can perform an abstract analysis in the case when f is leak-free.

One important question is whether this configuration leads to any benefit compared to simply performing f in the forward direction in both Mac and Ver, and comparing the tags using leakage-resilient value comparison. While this solution maybe indeed close, it has less flexibility as the security boils down to only the value comparison function: even if the PRI is perfectly secure, breaking the value comparison function using side channel analysis is sufficient to forge a tag. We would like to maintain the benefit of using the inverse function. In other words, we would like to maintain non-trivial security in the case when the PRI is safe but the value comparison is broken, but get better security when value comparison is secure. We also should point out that this is a generalization of the concept presented in [BGPS21], one can set $m = 0$ to remove reliance on value comparison completely. Another benefit of defining the value comparison function on $\mathcal{M}$ and not $\mathcal{C}$ is that depending on how much resources we are willing to allocate to the value comparison function, we can set $m \ll c$. Besides, our goal is to explore the design space using PRIs, including slightly worse combinations, rather than provide an optimal solution.

Leakage-Resilient Value Comparison Dobraunig and Mennink [DM21] defined the leakage-resilient value comparison model as follows: Consider a game where the challenger selects μ targets $\{M_1, \ldots, M_\mu\} \in \mathcal{M}^\mu$, the adversary makes q_d queries on the form (i, M') and the challenger returns `true` if an only if $M_i = M'$. It also returns the associated leakage function L_{VC}. Then, we say that VC is $(\mu, \varepsilon, q_d, t)$-$\mathsf{VC}$-secure if

$$\Pr[\{M_1, \ldots, M_\mu\} \xleftarrow{\$} \mathcal{M}^\mu : (i, M') \leftarrow \mathcal{A}^{\mathsf{VC}} | M' = M_i] \leq \varepsilon_{\mathsf{vc}}.$$

A special case is when we condition the probability such that the first q_d verification query return `false`. In other words,

$$\Pr[\{M_1, \ldots, M_\mu\} \xleftarrow{\$} \mathcal{M}^\mu : (i, M') \leftarrow \mathcal{A}^{\mathsf{VC}} | M' = M_i, \forall 1 \leq j \leq q_d, \mathsf{VC}(i_j, M'_j) = \texttt{false}]$$

$$\leq \varepsilon_{1-\mathsf{vc}},$$

in which case we say the scheme is $(\mu, \varepsilon_{\mathsf{vc}}, q_d, t)$-1-$\mathsf{VC}$-secure. To understand the relation between the two bound, consider $\mathcal{A}$ an adversary that terminates after the first successful forgery vs an adversary $\mathcal{B}$ that does not terminate. $\mathcal{A}$ simply runs $\mathcal{B}$ and terminates after the first successful forgery. Thus, both adversaries have the same advantage. By a simple hybrid argument, we can then see that

$$\varepsilon_{\mathsf{vc}} \leq q_d \varepsilon_{1-\mathsf{vc}}.$$

Next, we look at how to combine the security of the PRI in the leak-free model with the leakage-resilient VC. The main challenge is estimating μ. Note that if the adversary performs a query (K, H, M_1, C), followed by another query (K, H, M_2, C), $f^{-1}(K, H, C)$ is the same in both queries. However, the adversary does not trivially know whether $f^{-1}(K, H, C) = \perp$, or not. If $f^{-1}(K, H, C) = \perp$ then all such queries shall fail and the value comparison function does not operate on useful targets. Thus, μ should be the number of non-trivial unique values $f^{-1}(K, H, C) \neq \perp$ that appear in verification queries.

Proposition 3. *Let f be a $(\varepsilon_{\mathsf{pri}}, t_1)$-secure leak-free PRI defined according to Definition 2 and $\mathsf{VC} : (\{0,1\}^{\leq m} \cup \{\perp\}) \times \{0,1\}^{\leq m} \to \{\texttt{true}, \texttt{false}\}$ be a $(q_d, \varepsilon_{1-\mathsf{vc}}, q_d, t_2)$-1-$\mathsf{VC}$-secure value comparison function. Then, we can build a MAC $\mathsf{Mac} : \{0,1\}^k \times (\{0,1\}^h \times \mathcal{M}) \to \mathcal{C}$ such that Mac is $(\varepsilon_{\mathsf{sUF-L2}}, q_{\mathsf{L}}, q_e, q_d, t)$-sUF-L2 secure, with*

$$\varepsilon_{\mathsf{sUF-L2}} \leq \varepsilon_{\mathsf{pri}} + \frac{4 q_d \varepsilon_{1-\mathsf{vc}}}{2^{n-m}},$$

where $t_1 = \mathsf{lin}(t)$ and $t_2 = \mathsf{lin}(t)$, and

$$\mathsf{Mac}(K, H, M) = f(K, H, M)$$

and

$$\mathsf{Ver}(K, H, M, C) = \mathsf{VC}(f^{-1}(K, H, C), M).$$

Proof. We construct a sequence of hybrid games. Let E_i be the event that the adversary wins in game i. The adversary does not make trivial queries.
Game 0: the real-world game. The adversary wins if it forges the scheme.
Game 1: we replace the PRI with a $(\varepsilon_{\mathsf{pri}}, t_1)$-secure PRI.

$$|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_{\mathsf{pri}},$$

such that $t_1 = \mathsf{lin}(t)$.
Game 2: the game terminates if any query to VC returns `true`. First, we need to make sure that the oracle does not query VC with trivial queries, *i.e.*, queries that it trivially knows

the answer to. Consider if the adversary makes a verification query (H, M, C), followed by an encryption query (H, M'), such that $M \neq M'$ and $f(H, M') = C$. If such event occurs, the adversary trivially learns one of the targets of comparison. However, it cannot use this value to forge VC as the query becomes a trivial query for Ver. Thus, we can safely assume that for any query $\mathsf{Ver}(K, H, M, C)$, there is not previous query $\mathsf{Mac}(K, H, M) = C$. We can construct an adversary $\mathcal{B}$ that runs $\mathcal{A}$, observes the queries $\mathcal{A}$ makes and if the game terminates, it uses the last Ver query to break the value comparison game. $\mathcal{B}$ can index the value comparison targets using (H, C). In other words, there is an injective encoding from (H, C) to the target index space. However, $\mathcal{B}$ (or $\mathcal{A}$) can only succeed against the target indexed by (H, C) if $f^{-1}(K, H, C) \neq \perp$. Consider the PRI implemented using lazy sampling (Algorithm 1). The probability that this is true is determined by the first time $f^{-1}(K, H, C)$ is called, and bounded by

$$\frac{|\mathsf{EligX}|}{|\mathsf{EligY}|} \leq \frac{4}{2^{n-m}}.$$

We can construct a sequence of hybrid games such that game 1^0 is game 1 and game 1^{q_d} game 2. Game 1^i is the game that terminates if the i^{th} query is successful. We have that

$$|\Pr[E_{1^i}] - \Pr[E_{1^{i+1}}]| \leq \frac{4\varepsilon_{1-\mathsf{vc}}}{2^{n-m}}.$$

Thus,

$$|\Pr[E_1] - \Pr[E_2]| \leq \frac{4q_d\varepsilon_{1-\mathsf{vc}}}{2^{n-m}}.$$

If game 3 does not terminate, then $\mathcal{A}$ cannot win, *i.e.*, $\Pr[E_2] = 0$. The overall bound is given by

$$\Pr[E_2] + |\Pr[E_1] - \Pr[E_2]| + |\Pr[E_0] - \Pr[E_1]|.$$

□

Proposition 3 shows that we can recover some of the security using a leakage-resilient value comparison function. However, the value comparison function in the proof is assumed to handle as high as q_d targets. In practice, this is not tight as only the targets that are not $\perp$ are relevant to the adversary. On the other hand, if the value comparison function is leak-free, the bound becomes $O(q_d/2^n + \varepsilon_{\mathsf{pri}})$ which is almost optimal.

3.3 Instantiations and Practicalities

After we have shown the useful properties of PRIs, a valid question is how practical is this primitive. We look forward a little bit and consider how the the PRI will be used in the remainder of the paper. The most obvious way to instantiate a PRI is using a TBC in the EtE framework. If the TBC is an ideal TBC, then indeed the resulting PRI is almost ideal in two respects:

1. [BH24] proved the collision resistance of this approach, while we proved the collision resistance of an ideal PRI. The two bounds match, showing that PRIs built using EtE have the same collision resistance as ideal PRIs.
2. The ideal TBC keyed with a secret random key enjoys the *so-called* full STPRP security, which translates to full PRI security.

However, one has to be careful, as ideal TBC are not real-world primitives, and the way we know how to use constructions from ideal TBCs is by using adhoc TBC that have been built from scratch and have stood the test of time against cryptanalysis. Then, we assume that they behave as closely as possible to ideal TBCs. Typically, TBCs built from smaller primitives using security reductions are not designed to behave as ideal TBCs or be collision resistant. This presents a challenge, and opens avenues for new TBC designs as we discuss below.

PRIs with 64-bit collision resistance: One of our goals is building CMT4-secure AEAD scheme, which boils down to the collision resistance of the underlying PRI. In this scenario, collision resistance is the limiting factor, especially since the collision resistance can be foiled using only offline queries/time complexity of the adversary. However, we note that many practical *so-called* CMT4-secure AEAD schemes offer only 64-bit security: Ascon [DEMS21], whose CMT4 security was studied in [KSW23], and TEDT [BGP+20] and Triplex [SPS+22] AEAD schemes, whose CMT4 security was studied in [DEJ+24].

To match this security level, we can simply use a 128-bit TBC with large tweak space and set $m = 64$. Two such TBCs have been extensively studied: Deoxys-TBC [JNPS21] (winner of the CAESAR competition) and SKINNY [BJK+16] (finalist of the NIST lightweight cryptography project, as part of the Romulus family [GIK+21]). Using this set-up, with either TBCs, gives the same CMT4 security level as Ascon or the schemes in [DEJ+24], with only half the ciphertext expansion.

PRIs with >64-bit collision resistance: We believe we need higher security for CMT4 and collision resistance. [BH24] provides a PRI construction (HtM) that achieves PRI security and collision resistance. The collision resistance is sufficient (90-bit security using 2 calls to a 128-bit primitive). Its PRI security is studied with multi-keys and no tweaks. It is easily adaptable to our use case (single key, many tweaks). This is not an ideal PRI anymore, but it has dedicated analyses for collision resistance and PRI security. However, it suffers from a birthday-bound term in its PRI security.

We believe a 256-bit TBC is needed. However, there is no accepted scheme, to the best of our knowledge. There has been recent attempts: Pholkos [BLLS22] and Ghidle [NSA+23]. Whether these schemes stand the test of time remains to be seen.

Of course, another possibility is to use a BBB-secure domain extender of the TBC (HtM is arguably such an extender but with only birthday-bound security). Recent work on BBB TBC constructions focused on increasing the tweak size rather than the block size. Another alternative is to use a wide tweakable block cipher with BBB security, limiting its block size. However, such solutions are not very efficient for small messages and their collision resistance is not studied. Building a TBC domain extender that maintains collision resistance and does not suffer birthday bound drop, while being practical is an interesting problem.

All in all, our results can be seen as motivation to either study 256-bit TBCs or design efficient domain extenders.

A different instantiation of the PRI with value comparison using a forkcipher: A forkcipher is inherently a PRI. Its encryption function is a keyed function: $f : \{0,1\}^k \times \{0,1\}^h \times \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^n$. It takes a secret key K, tweak H and plaintext M and returns two ciphertext blocks C_l and C_r that are indistinguishable for two encryptions of M using two different TBCs. To invert the function, we need only one of the two ciphertext blocks. Given (K, H, C_l) we can recover (M, C_r) and given (K, H, C_r) we can recover (M, C_l). We can also have partial implementations that from one ciphertext block, generate only the other one or the plaintext. Consider a leak-free implementation of the forkcipher. Mac uses the forkcipher to generate the left ciphertext block $C_l = T$ from M. Ver generates C'_r form that T. It also uses the input plaintext M to generate a right ciphertext block C''_r and checks if $C'_r = C''_r$. If the forkcipher is ideal and leak-free, then the probability that any adversary find such a forgery should only be bounded by $O(q_d/2^n)$. A similar MAC is independently studied in [BSL24] but with smaller tweak space and without any part of the plaintext being processed by the forkcipher. This approach can be another interesting avenue for further studies.

4 A Leakage-Resilient Message Authentication Code

LRMAC1, depicted in Figure 1(a) is an elegant MAC construction proposed by Berti *et al.* [BGPS21] as a leakage-resilient MAC from non-idealized assumptions: strong unpredictability of the TBC and a collision-resistant public hash function. The designers assume that all the internal values of the hash function can be leaked, but the TBC remains unpredictable with leakage. The designers claim that it has beyond birthday security, particularly in the black-box model. In the previous section, we have shown an issue with this claim, such that if we would like to claim BBB security with known techniques, we need leak-freeness. In this section, we generalize LRMAC1 to iLRMAC (depicted in Figure 1(b)). Recently, Dhar *et al.* [DEJ+24] studied the collision resistance of LRMAC1, showing that it is collision-resistant up to half the block size of the TBC, as long as the hash function is collision-resistant and the TBC is an ideal cipher.

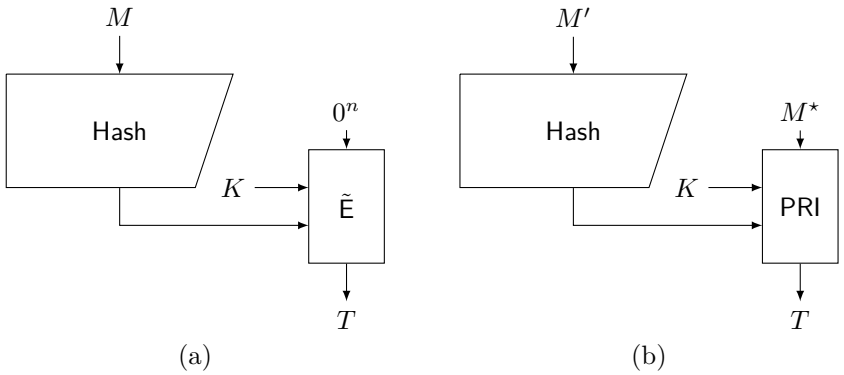

Figure 1: Leakage resilient MACs: (a) LRMAC1. (b) The newly proposed iLRMAC.

Generalized LRMAC1: The proposed MAC can be seen as a generalization of LRMAC1 [BGPS21] but allows more flexibility to adjust the security level based on the available implementations of heavily protected components; in this case, a keyed tweakable injective function and leakage-resilient value comparison.

The iLRMAC construction is depicted in Figure 1(b). Instead of a TBC, we use a PRI $f : \{0,1\}^k \times \{0,1\}^h \times \{0,1\}^{\leq m} \to \{0,1\}^n$, where $m < n$. If $|M| \leq m$, then $M^\star = M$ and $M' = \epsilon$ (the empty string). If $|M| > m$, then $M^\star = M[|M| - m + 1 : |M|]$ and $M' = M[1 : |M| - m]$. In the former case, $H_\epsilon = \mathsf{Hash}(M')$ is the hash tag of the empty string, and the tag is computed as $f(K, H_\epsilon, M^\star)$. Since H_ϵ can be precomputed, the cost when the message length is less than m is just one call to the f.

4.1 Collision Resistance

In this paragraph, we study the collision resistance of the MAC construction described in Algorithm 2.

Theorem 1. *Let $f : \mathcal{K} \times \{0,1\}^h \times \{0,1\}^{\leq m} \to \{0,1\}^n$ be a (ε_1, t_1)-collision-resistant PRI, such that $n > m$ and $f^{-1} : \{0,1\}^k \times \{0,1\}^h \times \{0,1\}^n \to \{0,1\}^{\leq m} \cup \{\perp\}$ is its inverse. Let $H : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^h$ be a (ε_2, t_2)-collision-resistant hash function. Then, iLRMAC is (ε, t)-collision-resistant, where*

$$\varepsilon \leq \varepsilon_1 + \varepsilon_2,$$

where $t_1 = \mathsf{lin}(t)$ and $t_2 = \mathsf{lin}(t)$.

Proof. Suppose $\mathcal{A}$ outputs (K_1, M_1) and (K_2, M_2) such that $\mathsf{iLRMAC}[H, f](K_1, M_1) = \mathsf{iLRMAC}[H, f](K_2, M_2)$.

Let $\mathcal{B}$ be a collision finding adversary against H and $\mathcal{C}$ be a collision finding adversary against f. $\mathcal{B}$ runs $\mathcal{A}$ and outputs (M_1', M_2'). We use a hybrid argument: let game 0 be the collision resistance game, and game 1 terminates if $\mathcal{B}$ is successful. Let E_i be the event that $\mathcal{A}$ wins in game i. Thus,

$$|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_2.$$

Let $\mathcal{C}$ be an adversary trying to find a collision for f that runs $\mathcal{B}$, and let game 2 be almost the same as game 1, but $\mathcal{B}$ returns $(M_1', M_2', H(M_1'), H(M_2'), K_1, K_2, M_1^\star, , M_2^\star)$ instead of just (M_1', M_2'). $\mathcal{C}$ returns $((K_1, H(M_1'), M_1^\star), (K_2, H(M_2'), M_2^\star))$. This can only improve $\mathcal{C}$'s advantage;

$$\Pr[E_1] \leq \Pr[E_2].$$

Besides, if the game does not terminate, then $H(M_1') \neq H(M_1')$ or $M_1' = M_2'$. The latter implies $(K_1, M_1^\star) \neq (K_2, M_2^\star)$. $\mathcal{A}$ can win only if either $\mathcal{B}$ (in which case the game would terminate) or $\mathcal{C}$ win. Thus,

$$\Pr[E_2] \leq \varepsilon_1.$$

The overall bound is

$$\Pr[E_1] + |\Pr[E_0] - \Pr[E_1]| \leq \Pr[E_2] + |\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_1 + \varepsilon_2.$$

We note that the running time of both $\mathcal{B}$ and $\mathcal{C}$ is the running time of $\mathcal{A}$ in addition to a constant number of checks. □

4.2 strong Unforgeability with Leakage

The LRMAC1 construction [BGPS21] can be retrospectively seen as a special case of our iLRMAC, where $M^\star = \epsilon$ and the PRI is implemented using EtE. Thus, we expect the security properties of LRMAC1 to also generalize to iLRMAC. The iLRMAC is formally depicted in Algorithm 2.

Algorithm 2 The iLRMAC construction with leakage-resilient value comparison. L_e and L_d are according to Definition 7.

1: $\mathsf{Mac}(K, M)$:	5: $\mathsf{Ver}(K, M, T)$:
2: $M^\star, M' \leftarrow \mathtt{parse}(M)$	6: $M^\star, M' \leftarrow \mathtt{parse}(M)$
3: $H = \mathsf{Hash}(M')$	7: $H = \mathsf{Hash}(M')$
4: **return** $\mathsf{L}_e(H, M)$	8: **return** $\mathsf{L}_d(H, M, C)$

Theorem 2. *Let H be a $(\varepsilon_{\mathsf{cr}}, t_1)$-collision resistant hash function. Let L_e and L_d be two oracles according to Definition 7 where the short MAC is $(q_L, q_e, q_d, t_2, \varepsilon_{\mathsf{cr}} + \varepsilon'_{\mathsf{sUF-L2}})$-sUF-L2-secure. Then, iLRMAC is $(q_e, q_d, t, \varepsilon_{\mathsf{cr}} + \varepsilon'_{\mathsf{sUF-L2}})$-sUF-L2 with:*

$$\varepsilon_{\mathsf{sUF-L2}} \leq \varepsilon_{\mathsf{cr}} + \varepsilon'_{\mathsf{sUF-L2}},$$

with $\mathsf{t}_1 = \mathsf{lin}(t)$ and $\mathsf{t}_2 = \mathsf{lin}(t)$.

Proof. The proof follows from a simple hybrid argument, where we consider a hybrid game that terminates if a collision is found for the hash. If no collision exists, then the adversary can only win if it forges the fixed length MAC. □

Our results show that we can add more flexibility to LRMAC1 by processing part of the plaintext by the heavily protected component, at the cost of using a leakage-resilient value comparison function. However, the value comparison function does not have to be very costly. Dobraunig and Mennink [DM21] showed that this function does not have to be heavily protected if implemented using a public primitive. Thus, we believe iLRMAC is a valuable generalization of LRMAC1 adding more space for flexibility and trade-offs. The system engineer can decide how much weight to put on the leakage-based adversaries. The system can have strong black box security, and weaker, but non-trivial, leakage-resilient security.

5 Succinctly-Committing Online AEAD

In this section, we use PRIs to construct a succinctly-committing AEAD scheme with CIML2 security and black-box privacy. The scheme can also provide privacy with leakage, but we leave this out of scope as standard techniques can be used to instantiate our construction. Instead, we focus on this construction as a blueprint, similar to the blueprints discussed in [DEJ+24]. We use a function that has partial collision resistance: The function takes almost all the inputs of the AEAD scheme, it generates the ciphertext and a hashed value of the ciphertext. For a fixed key, the hashed value is collision-resistant. This is formalized in Section 2.

Note that H is not a hash function: we make no claims about its collision resistance for different keys. It is also neither a committing AEAD scheme nor an AEAD scheme in general, as we make no claims about the randomness of the hashed value. Such function could have many realizations. It could be realized a duplex sponge construction without squeezing. It could also be realized as a stream cipher keyed with $K \in \mathcal{K}$, followed by a public hash function of the ciphertext, nonce and associated data. Triplex [SPS+22] without finalization, is also a realization of this function. This way, we can make general claims about a wide class of realizations. We define an AEAD scheme as follows:

Definition 8. Let $H : \mathcal{K}_h \times \mathcal{K} \times \mathcal{N} \times \mathcal{A} \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^h$ be an (ε_1, t_1)-collision-resistant encryption function. Let $f : \mathcal{K} \times \{0,1\}^h \times \{0,1\}^{\leq m} \to \{0,1\}^n$ be an (ε_2, t_2)-collision-resistant PRI. Then, $\Pi[H, \tilde{\pi}]$ is an AEAD scheme defined in Algorithm 3. The ciphertext expansion of Π is $n - m$ for messages longer than m bits.

Algorithm 3 The scoAE construction.

$\Pi.\mathsf{Enc}(K, N, A, M)$:
$M^\star, M' \leftarrow \mathtt{parse}(M)$
$(C, V) \leftarrow H(K, N, A, M')$
return $(C, f(K, V, M^\star))$

$\Pi.\mathsf{Dec}(K, N, A, C, T)$:
$(M', V) \leftarrow H^{-1}(K, N, A, C)$
$M^\star = f^{-1}(K, V, T)$
if $M^\star = \perp$ **then**
  return $\perp$
else
  return $M' \| M^\star$
end if

Theorem 3. *Let $\Pi[H, f]$ be the AEAD scheme given in Definition 8. Let H be an (ε_1, t_1)-collision-resistant encryption function and f be an (ε_2, t_2)-collision-resistant PRI. Then,* Pi *is $(\varepsilon_{\mathsf{cmt4}}, t)$-CMT4 secure, such that*

$$\varepsilon_{\mathsf{cmt4}} \leq \varepsilon_1 + \varepsilon_2,$$

where $t_1 = \mathsf{lin}(t)$ and $t_2 = \mathsf{lin}(t)$.

Proof. Let $\mathcal{A}$ be a CMT4 adversary that outputs

$$(K_1, N_1, A_1, M_1, K_2, N_2, A_2, M_2).$$

Let $\mathcal{B}$ be a collision finding adversary against H that runs $\mathcal{A}$ and outputs

$$(K_1, N_1, A_1, M_1', N_2, A_2, M_2')$$

if $K_1 = K_2$. Let $\mathcal{C}$ be collision-finding adversary against f that runs $\mathcal{B}$ and has access to the output of H, similar to the proof of Theorem 1. It outputs $((K_1, V_1, M_1^\star), (K_2, V_2, M_2^\star))$ We consider two types of challenges that $\mathcal{A}$ may return:

1. $\underline{K_1 = K_2}$: In this case, if $(K_1, V_1) = (K_2, V_2)$, then a collision on the tag occurs only if $M_1^\star = M_2^\star$. If $M_1^\star = M_2^\star$, then it must hold that $(N_1, A_1, M_1') \neq (N_2, A_2, M_2')$ for the challenge to be valid, and if $\mathcal{A}$ is successful, then $\mathcal{B}$ is successful. If $M_1^\star \neq M_2^\star$ and $V_1 = V_2$, then $\mathcal{A}$ cannot be successful. If If $M_1^\star \neq M_2^\star$ and $V_1 \neq V_2$, then if $\mathcal{A}$ is successful, then $\mathcal{C}$ is successful: $\varepsilon_{\mathsf{cmt4}} \leq \varepsilon_1 + \varepsilon_2$.

2. $\underline{K_1 \neq K_2}$: In this case, if $\mathcal{A}$ is successful, then $\mathcal{C}$ must be successful: $\varepsilon_{\mathsf{cmt4}} \leq \varepsilon_2$.

The bound follows from the maximum of the two cases. □

Next, we study the privacy of scoAE. Since the details of of H are abstracted away, we cannot argue its security in the presence of leakage in a meaningful way, and limit our analysis to the black-box setting. Standard techniques, such as those used in TEDT or Triplex, can be used to elevate the result to the leakage resilience setting. Note that for CMT4 security, we assumed ideal primitives, but for privacy and integrity, this is note always the case. We face an issue where both H and f are keyed by the same key, which is needed for CMT4 security. In order to make sure the appropriate domain separation is maintained, we need to go one level of abstraction lower, and define how H processes the key, relative to f. This is done in Definition 9.

Definition 9. Let $H : \mathcal{K}_h \times \{0,1\}^n \times \mathcal{N} \times \mathcal{A} \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^h$ be an (ε_1, t_1)-collision-resistant encryption function with unbounded leakage. Let $\tilde{E} : \mathcal{K} \times \{0,1\} \times \{0,1\}^h \times \{0,1\}^n \to \{0,1\}^n$ be a leak-free TBC. Then, $\Pi[H, \tilde{E}]$ is an AEAD scheme defined in Algorithm 4. The ciphertext expansion of Π is $n - m$ for messages longer than m bits.

Note that in this construction, the second call to the TBC is simply a PRI implemented using EtE. For privacy, we also need an assumption on the pseudo-randomness of H. In particular, for the purposes of the privacy security proof, we need to assume that H is a random oracle. This may seem inconvenient, but we recall that our privacy proof is not for a particular scheme but for a generic high level blueprint, and once H is fixed to a specific function, it is very likely that this assumption can be removed. One may argue about the value of such proof, but we believe this is necessary to show the soundness of the scheme overall.

Definition 10. Let $H : \{0,1\}^n \times \mathcal{N} \times \mathcal{A} \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^h$ be an idealized encryption function with a auxiliary output. We assume H behaves a random oracle.

Theorem 4. *Let $\Pi[H, \tilde{E}]$ be the AEAD scheme given in Definition 8. Let H be a random oracle with the interface in Definition 10 and $\tilde{E}$ be an $(2q_e + 2q_d, t_1, \varepsilon_{\mathsf{stprp}})$-secure TBC. Then for any nonce-respecting adversary $\mathcal{A}$ against the IND-CPA security of $\Pi[H, f]$, making q_e queries to $\Pi.\mathsf{Enc}$ and q_H queries to H and running in time at most t,*

$$\mathrm{Adv}^{\mathsf{indcpa}}_{\Pi}(\mathcal{A}) \leq \varepsilon_{\mathsf{stprp}} + \frac{q_e^2 + q_H^2 + q_e q_H}{2^h} + \frac{q_H}{2^n} + \frac{2q_e}{2^{n-m}}$$

where $t_1 = \mathsf{lin}(t)$.

Algorithm 4 The scoAE construction with a heavily protected Key derivation function.

$\Pi.\mathsf{Enc}(K, N, A, M)$:
$K_0 \leftarrow \tilde{E}(K, 0, 0^h, N)$
$M^\star, M' \leftarrow \mathtt{parse}(M)$
$(C, V) \leftarrow H(K_0, N, A, M')$
return $C \| \tilde{E}(K, 1, V, M^\star \| 1 \| 0^{n-|M|-1})$

$\Pi.\mathsf{Dec}(K, N, A, C, T)$:
$K_0 \leftarrow \tilde{E}(K, 0, 0^h, N)$
$(M', V) \leftarrow H^{-1}(K_0, N, A, C)$
$M^\star = (\tilde{E}^{-1})(K, 1, V, C)$
if $M^\star = \perp$ **then**
  return $\perp$
else
  return $M' \| M^\star$
end if

Proof. The adversary has access to $\Pi.\mathsf{Enc}$. It also has access to H and can make queries to it with chosen inputs. We construct a sequence of hybrid games. Let E_i be the event that the adversary wins in game i.
Game 0: the real-world game.
Game 1: We replace the TBC with a uniformly random family of tweakable permutations $\tilde{\pi}$:

$$|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_{\mathsf{stprp}}.$$

Game 2: We terminate the game if the adversary makes a query to H directly with inputs (K_0, N, A_1, M_1') and a query to $\Pi.\mathsf{Enc}$ with input $(N, A_2, M_2' \| M_2^\star)$, such that $\tilde{\pi}(0, 0^h, N) = K_0$. Since the adversary is nonce-respecting, each call to $\Pi.\mathsf{Enc}$ has a unique nonce. For the i^{th} query, the probability of this event is bounded by

$$\frac{q_H^{N_i}}{2^n},$$

where $q_H^{N_i}$ is the number of queries to H on the form $(\cdot, N_i, \cdot, \cdot)$. We sum over all queries to get

$$|\Pr[E_1] - \Pr[E_2]| \leq \frac{\sum_{i=1}^{q_e} q_H^{N_i}}{2^n} = \frac{q_H}{2^n}.$$

Game 3: We construct an adversary $\mathcal{C}$ against the the collision resistance of H. At the end of the game, we reveal all the auxiliary outputs of H. $\mathcal{C}$ runs $\mathcal{A}$. After the last query and the reveal of auxiliary outputs during queries to $\Pi.\mathsf{Enc}$. $\mathcal{C}$ also records the queries made directly to H. $\mathcal{C}$ checks the augmented transcript and if there is a collision in any of the implicit or explicit H queries. Since we are in the single key setting, this can only happen if a collision occurs in H. Thus, from the random oracle assumption,

$$|\Pr[E_2] - \Pr[E_3]| \leq \frac{q_e^2 + q_H^2 + q_e q_H}{2^h},$$

and the time of $\mathcal{C}$ is the time of $\mathcal{A}$, the time needed to reveal the auxiliary outputs and q_e checks.
Game 4: we replace the final TBC call by a random function F. Note that the first bit of the tweak ensures the permutations sampled at this step are independent of the permutation sampled during the first call. This transition is akin to a restricted PRP-PRF switch. Consider the tweakable permutation is implemented using lazy sampling: first T is sampled uniformly at random, and if the sampled block has appeared as an output for any $F(V_j, \cdot)$ call, it is resampled appropriately.

For a query $j \in \{1, \ldots, q_e\}$ to F: $F(V_j, M_j^\star)$, it always returns a random block unless that block has appeared for any $F(V_j, \cdot)$ call. Since game 2 would have terminated if a collision on V_j has occurred, we can assume that if $V_j = V_i$, then $(N_j, A_j, M_j') = (N_i, A_i, M_i')$. The number of candidates for such collision is bounded by $\min(q_e, 2^{m+1} - 2)$, since the adversary can make at most $2^{m+1} - 1$ such queries. Thus, the probability of this

collision is at most $\min(q_e/2^n, (2^{m+1}-2)/2^n)$. We take the union bound over q_e queries, we get

$$|\Pr[E_3] - \Pr[E_4]| \leq \frac{\min(q_e^2, (2^{m+1}-2)q_e)}{2^n} \leq \frac{2q_e}{2^{n-m}}.$$

Given no collisions occur on V, and the second TBC call is now replaced by a random function, the tag T is indistinguishable from random in all queries. Thus, the only way $\mathcal{A}$ can distinguish the ciphertexts generated by Π from random is by distinguishing the ciphertext output of H from random. Since H is a random oracle, and its inputs during $\Pi.\mathsf{Enc}$ do not repeat ($\mathcal{A}$ is nonce-respecting):

$$\Pr[E_4] = 0.$$

The final bound is then

$$\sum_{i=0}^{3} |\Pr[E_i] - \Pr[E_{i+1}]| \leq \varepsilon_{\mathsf{stprp}} + \frac{q_e^2 + q_H^2 + q_e q_H}{2^h} + \frac{q_H}{2^n} + \frac{2q_e}{2^{n-m}}.$$

□

Next, we show the CIML2 security. In this case, we do not need the random oracle assumption, and the proof is a lot simpler.

Theorem 5. *Let $\Pi[H, \tilde{E}]$ be the AEAD scheme given in Definition 9. Let H be an $(\varepsilon_{\mathsf{cr}}, t_1)$-strongly-collision-resistant encryption function, $\tilde{E}$ be a $(2q_e + 2q_d, t_2, \varepsilon_{\mathsf{stprp}})$-leak-free TBC. Then, for any adversary $\mathcal{A}$ against the CIML2 security of Π that makes q_e encryption queries and q_d decryption queries and runs in time t, there exists an adversary $\mathcal{B}$ against the STPRP security of $\tilde{\pi}$ which makes $2q_e + q_d$ forward queries and q_d backward queries, and an adversary $\mathcal{C}$ against the collision resistance of H such that*

$$\mathrm{Adv}^{\mathsf{ciml2}}_{\Pi}(\mathcal{A}) \leq \varepsilon_{\mathsf{stprp}} + \varepsilon_{\mathsf{cr}} + \frac{4q_v}{2^{n-m}},$$

where $t_1 = \mathsf{lin}(t)$ and $t_2 = \mathsf{lin}(t)$.

Proof. First, we replace $\tilde{E}$ with a tweakable uniformly random permutation (TURP). This gives the first term of the bound. This gives the first term of the bound.

Next, we construct an adversary $\mathcal{C}$ that has access to the TURP and H and simulates Π. It records all the queries made to H and terminates the game if a collision is found such that $(C_1, V_1) = H(K_{0,1}, A_1, M_1')$, $(C_2, V_2) = H(K_{0,2}, A_2, M_2')$, $(K_{0,1}, A_1, M_1') \neq (K_{0,2}, A_2, M_2')$ and $V_1 = V_2$. Let game 1 be the game where the adversary interacts with Π and game 2 be the game where the adversary interacts with $\mathcal{C}$, where E_i is the probability that the adversary wins in game i. Then,

$$|\Pr[E_1] - \Pr[E_2]| \leq \varepsilon_{\mathsf{cr}}.$$

Next, we consider the PRI built by the second call to the TURP. As discussed in Section 3, it can be seen as a fixed-length MAC. We now consider an adversary $\mathcal{B}$ trying to forge the fixed length MAC. In order to use $\mathcal{C}$ in a security reduction, we need to show that $\mathcal{C}$ does not make trivial queries to the fixed-length MAC. That is; it never queries the forward direction with the same (V, M) twice, does not call the backward direction with queries it knows the response to and does not attempt the same forgery (V, T) more than once. Since there is no collisions on V, then each V that appears during encryption or decryption queries of the AEAD corresponds to a unique tuple (K_0, N, A, M', C). During encryption, the adversary may attempt the same tuple multiple times but has to change $M^\star$, since they do not make trivial AEAD queries. Similarly, during decryption if they

ask for the decryption of $(N, A, C, \cdot)$ multiple times, they have to make the fourth input unique in all these attempts. Lastly, if the adversary attempts a forgery that triggers a backward call to the PRI with input (V, T), where (V, T) has appeared in a previous encryption query, then either the query is a trivial AEAD query or there is a collision on V. Thus, we can describe the reduction as follows: $\mathcal{B}$ runs $\mathcal{C}$. If $\mathcal{C}$ does not terminate then all the calls it makes to the fixed-length MAC are non-trivial queries. If $\mathcal{A}$ (which $\mathcal{C}$ runs) outputs a success forgery, then $\mathcal{B}$ outputs the corresponding (V, T). Note that the adversaries can observe V as H has unbounded leakage. In the leak-free model of the TBC, the probability of a successful forgery is bounded by

$$\Pr[E_2] \leq \frac{4q_d}{2^{n-m}}.$$

This concludes the proof. □

Comments on Theorems 3, 4 and 5: scoAE is generic enough that it allows us to cover a wide class of AEAD schemes, while also being specific enough to allow us to make significant security claims. Consider a message of length $l > m$ bits. Then, scoAE generates a ciphertext of length $l - m$ and an n-bit tag. Thus, the ciphertext expansion is $(l - m + n) - l = n - m$. It provides CMT4 security up to the collision resistance of PRI (typically, it is easier to find collisions for the PRI than for the hash function). If the PRI is implemented using a dedicated 128-bit TBC modeled as a leak-free ideal TBC, and $m = 64$, then scoAE provides 64-bit CIML2 security and 64-bit CMT4 security, using 64-bit expansion. This is optimal, and applies also with nonce misuse and leakage. To put this into perspective, Dhar *et al.* showed that TEDT [BGP+20] provides n-bit CIML2 security and $n/2$-bit CMT4 security. Instantiated with the same TBC, TEDT provides 128-bit CIML2 security and 64-bit CMT4 security with 128-bit expansion. If we instantiate TEDT with a 64-bit TBC, we get similar CIML2 security to that of scoAE but only half the CMT4 security. Besides, we could instantiate the PRI using two TBC calls, using HtM, getting very good CMT4 and CIML2 security, as 64-bit CMT4 security can be considered insufficient.

The security proof for privacy is admittedly tedious and requires a strong assumption on H. However, we find this to be natural, and a function of our high abstraction level. Since H is not defined at a low level, it is natural than we need to make such a strong assumption. We emphasize that the random oracle assumption is only used for privacy and is not used to CMT4 or CIML2 security.

We note that scoAE provides a blueprint for enhancing the security of existing committing AEAD schemes with minimal changes. scoAE in its most abstract form (Theorem 3) can be seen as the three phases of a duplex sponge design, such as Ascon, where the initialization and absorption parts represent H and the squeezing/finalization part is replaces by PRI. The same can be said about Encrypt-then-MAC schemes, *e.g.* TEDT, and Triplex.

6 Succinctly-Committing Misuse-Resistant AE

In this section, we present a second application of PRIs in the design of succinctly-committing AEAD. However, we consider misuse-resistant AEAD, in a construction dubbed scMRAE. The scheme is depicted in Figure 2 and described in details in Algorithm 5. Before we delve into the security proofs, we provide some intuition. In terms of commitment, it is easy to see that the security boils down to the collision-resistant of the hash-then-PRI part of the construction. For AEAD security, consider an adversary making encryption queries and no collision is found for the hash function. Thus, in all the encryption queries, each triplet (N, A, M') maps to a unique hash value. We can use a PRP-PRF-switch argument

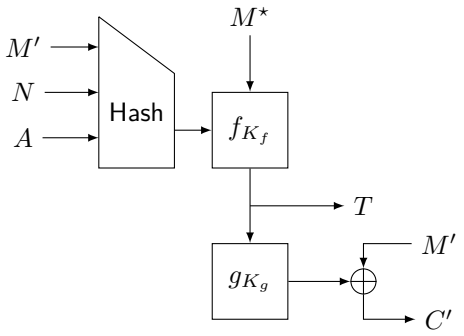

Figure 2: The scMRAE construction.

to show that when the tweak repeats only a small number of times, the output of the PRI is indistinguishable from random. Let μ be the number of times the triplet (N, A, M') appears in in encryption queries, then f_K can be viewed as a PRF with a logarithm security degradation μ. In Figure 2, g_K is a stream cipher with T as an IV. As long as T does not repeat, then the stream cipher is secure. If (N, M') repeats in two queries, then either A or $M^\star$ must be different. If A is the also repeated in the same queries, then T cannot be repeated, while if $M^\star$ is repeated, then A (and by assumption the hash value) cannot be repeated. Thus, the security boils down to the collision resistance of f_K. We note that for commitment, we require that the keys of f and g use exactly the same key (or dependent keys [DEJ+24]). However, this leads to an issue for MRAE security notions, which leads to an unnatural assumption; that it is secure to use the same key in both f and g. Fortunately, there are multiple ways to satisfy this assumption. One could build f and g from TBCs with disjoint tweak domains (using domain separation), or could use the pseudo-random number generator to generate two dependent keys from one key, in which case the pseudo-random number generator can be seen as part of f and g. We use the latter approach.

Algorithm 5 The scMRAE construction.

$\Pi.\mathsf{Enc}(K, N, A, M)$:
$(K_f, K_g) \leftarrow \mathsf{PRNG}(K)$
$M^\star, M' \leftarrow \mathtt{parse}(M)$
$V \leftarrow H(N, A, M')$
$T \leftarrow f(K_f, V, M^\star)$
$C' \leftarrow g_{K_g}(T) \oplus M'$
return $C' \| T$

$\Pi.\mathsf{Dec}(K, N, A, C, T)$:
$(K_f, K_g) \leftarrow \mathsf{PRNG}(K)$
$M' \leftarrow g_{K_g}(T) \oplus C'$
$V \leftarrow H(N, A, M')$
$M^\star \leftarrow f^{-1}(K_f, V, T)$
if $M^\star = \perp$ **then**
  return $\perp$
else
  return $M' \| M^\star$
end if

Before studying the scheme we need to define the properties of PRNG.

Definition 11. Let $\mathsf{PRNG} : \mathcal{K} \to \mathcal{K}^2$ be a pseudo-random number generator. We say PRNG is a (ε, t)-secure pseudo-random number generator if for any adversary $\mathcal{A}$ running in time at most t:

$$|\Pr[K \xleftarrow{\$} \mathcal{K} : 1 \leftarrow \mathcal{A}^{\mathsf{PRNG}}] - \Pr[1 \leftarrow \mathcal{A}^{\$}]| \leq \varepsilon,$$

where \$ is a random oracle that returns two random values in $\mathcal{K}^2$.

Definition 12. Let $\mathsf{PRNG} : \mathcal{K} \to \mathcal{K}^2$ be a pseudo-random number generator. We say PRNG is a (ε, t)-collision resistant pseudo-random number generator if for any adversary

$\mathcal{A}$ running in time at most t:

$$\Pr[(K_{f_1}, K_{g_1}, K_{f_2}, K_{g_2}) \leftarrow \mathcal{A}\ s.t. K_{g_1} \neq K_{g_2} \wedge K_{f_1} = K_{f_2}] \leq \varepsilon$$

and

$$\Pr[(K_{f_1}, K_{g_1}, K_{f_2}, K_{g_2}) \leftarrow \mathcal{A}\ s.t. K_{f_1} \neq K_{f_2} \wedge K_{g_1} = K_{g_2}] \leq \varepsilon.$$

Theorem 6. *Let $H : \mathcal{K}_h \times \{0,1\}^* \times \mathcal{N} \times \mathcal{A} \rightarrow \{0,1\}^h$ be an (ε_1, t_1)-collision-resistant hash function. Let $f : \mathcal{K} \times \{0,1\}^{\leq m} \times \{0,1\}^h \times \{0,1\}^n$ be a (ε_2, t_2)-collision-resistant PRI. Let $g : \mathcal{K} \times \{0,1\}^h \rightarrow \{0,1\}^*$ be a pseudo-random number generator. Let $\mathsf{PRNG} : \mathcal{K} \rightarrow \mathcal{K}^2$ be a (ε_3, t_3)-collision-resistant pseudo-random number generator. Let $\Pi[H, f, g]$ be the authenticated encryption scheme where the encryption algorithm is defined in Algorithm 5. Then, $\Pi[H, f, g]$ is $(\varepsilon_{\mathsf{cmt4}}, t)$-CMT4 secure AEAD scheme, with*

$$\varepsilon_{\mathsf{cmt4}} \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3,$$

$t_1 = \mathsf{lin}(t)$, $t_2 = \mathsf{lin}(t)$ and $t_3 = \mathsf{lin}(t)$.

Proof. Let $\mathcal{A}$ be a CMT4 adversary that output $(K_1, N_1, A_1, M_1), (K_2, N_2, A_2, M_2)$. First, consider the case that $(K_{g_1}, C_1') \neq (K_{g_2}, C_2')$ and $M_1' = M_2'$. In this case, the adversary can trivially win if $K_{f_1} = K_{f_2}$. The probability that the adversary finds key pairs (K_{f_1}, K_{g_1}) and (K_{f_2}, K_{g_2}) that satisfy this condition is upper bounded by ε_3. Next we assume that $K_{g_1} \neq K_{g_2}$ implies $K_{f_1} \neq K_{f_2}$ and vice versa. If $K_{g_1} = K_{g_2}$, then we can see that $C_1' = C'2$ implies $M_1' = M_2'$. Thus, it must hold that $(K_{f_1}, N_1, A_1, M_1^\star) \neq (K_{f_2}, N_2, A_2, M_2^\star)$. If $K_{g_1} \neq K_{g_2}$ and $K_{f_1} \neq K_{f_2}$, then also $(K_{f_1}, N_1, A_1, M_1^\star) \neq (K_{f_2}, N_2, A_2, M_2^\star)$. In both cases, the inputs to the hash function and f are not equal. We analyze this case below.

We note that the challenge can only be successful if there is a collision on the tag T. Consider an adversary $\mathcal{B}$ that tries to find a collision against H. It runs $\mathcal{A}$ and outputs $((N_1, A_1, M_1'), (N_2, A_2, M_2'))$. If $(N_1, A_1, M_1') = (N_2, A_2, M_2')$, then $\mathcal{B}$ cannot win. If $(N_1, A_1, M_1') \neq (N_2, A_2, M_2')$ then $\mathcal{B}$ wins if $H(N_1, A_1, M_1') = H(N_2, A_2, M_2')$. Let E_0 be the event that $\mathcal{A}$ wins in the original CMT4 game, and E_1 be the event that $\mathcal{A}$ wins if $\mathcal{B}$ does not win. Then,

$$|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_1.$$

On the other hand, if $\mathcal{B}$ does not win, $H(N_1, A_1, M_1') \neq H(N_2, A_2, M_2')$. and $\mathcal{A}$ can only win if a collision for f is found.

$$\Pr[E_1] \leq \varepsilon_2.$$

□

Next, we look at integrity. We reduce the security to the MAC in Theorem 2, using a trick proposed in [IKMP20], where we give the adversary access to the MAC and stream cipher separately. $\mathsf{int-ctxt}$ and $\mathsf{ind-cpa}$ refer to the integrity and confidentiality security notions defined in Section 2, with nonce misuse but without leakage.

Theorem 7. *Let $H : \mathcal{K}_h \times \{0,1\}^* \times \mathcal{N} \times \mathcal{A} \rightarrow \{0,1\}^h$ be an (ε_1, t_1)-collision-resistant hash function. Let $f : \mathcal{K} \times \{0,1\}^{\leq m} \times \{0,1\}^h \times \{0,1\}^n$ be a (ε_2, t_2)-secure PRI according to Definition 2. Let $g : \mathcal{K} \times \{0,1\}^h \rightarrow \{0,1\}^*$ be a (ε_3, t_3)-secure pseudo-random number generator. Let $\mathsf{PRNG} : \mathcal{K} \rightarrow \mathcal{K}^2$ be a (ε_4, t_4)-secure pseudo-random number generator. Let $\Pi[H, f, g]$ be the authenticated encryption scheme where the encryption algorithm is defined in Algorithm 5. Then, for any adversary $\mathcal{A}$ making q_e encryption queries and q_d decryption queries and running in time at most t,*

$$\mathrm{Adv}_{\Pi}^{\mathsf{int-ctxt}}(\mathcal{A}) \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \varepsilon_4 + \frac{4q_d}{2^{n-m}},$$

$t_1 = \mathsf{lin}(t)$, $t_2 = \mathsf{lin}(t)$, $t_3 = \mathsf{lin}(t)$ and $t_4 = \mathsf{lin}(t)$.

Proof. First, (K_f, K_g) by two uniformly random keys. This gives the term ε_4. Second, we replace f and g with a uniformly random tweakable injection and a uniform random number generator g', respectively. This gives the terms ε_2 and ε_3 of the overall bound. Next, we modify the game by allowing the adversary access to the random number generator where the adversary makes queries to the MAC

$$T = f'(H(M', N, A), M^\star),$$

and the random number generator g' separately, and is required to win a forgery game against the MAC. This can only improve the adversary's advantage and reduces the security to Theorem 2 with leak-free PRI and no value comparison, which give the bound

$$\varepsilon_1 + \frac{4q_d}{2^{n-m}}.$$

Combining both bounds gives the overall bound. □

Last but not least, we look at the confidentiality of the scheme.

Theorem 8. *Let $H : \mathcal{K}_h \times \{0,1\}^* \times \mathcal{N} \times \mathcal{A} \to \{0,1\}^h$ be an (ε_1, t_1)-collision-resistant hash function. Let $f : \mathcal{K} \times \{0,1\}^{\leq m} \times \{0,1\}^h \times \{0,1\}^n$ be a (ε_2, t_2)-secure PRI according to Definition 2. Let $g : \mathcal{K} \times \{0,1\}^h \to \{0,1\}^*$ be a (ε_3, t_3)-secure pseudo-random number generator. Let $\mathsf{PRNG} : \mathcal{K} \to \mathcal{K}^2$ be a (ε_4, t_4)-secure pseudo-random number generator. Let $\Pi[H, f, g]$ be the authenticated encryption scheme where the encryption algorithm is defined in Algorithm 5. We assume that f and g can be securely keyed by the same key without impacting their respective security. Then, for any adversary that makes q_e encryption queries and runs in time at most t,*

$$\mathrm{Adv}^{\mathsf{ind-cpa}}_{\Pi}(\mathcal{A}) \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \varepsilon_4 + \frac{(\mu - 1)q_e}{2^n} + \frac{\binom{q_e}{2}}{2^n},$$

where each triplet (N, A, M') is repeated at most $\mu \leq 2^{m+1} - 1$ times, $t_1 = \mathsf{lin}(t)$, $t_2 = \mathsf{lin}(t)$ and $t_3 = \mathsf{lin}(t)$.

Proof. First, (K_f, K_g) by two uniformly random keys. This gives the term ε_4. Second, we replace f and g with a uniformly random tweakable injection and a uniform random number generator g', respectively. This gives the terms ε_2 and ε_3 of the overall bound. Next, we shall define a sequence of hybrid games, where E_i is the event that the adversary is able to distinguish the ciphertexts from random strings in game i.
Game 0: the original game with f' and g'.
Game 1: We consider an adversary $\mathcal{B}$ that has access to H, f' and g'. It simulates the scheme and responds to $\mathcal{A}$'s encryption queries. It records all the queries made to H and terminates the game if a collision is found. Thus,

$$|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_1.$$

Game 2: In this game, f' is replace by a random function. This transition is akin to a restricted PRP-PRF switch. Consider f' is implemented using lazy sampling: during the j^{th} query, T_j is first sampled uniformly at random, and if the sampled block has appeared as an output for any $f'(V_j, \cdot)$ call, it is resampled appropriately.

For a query $j \in \{1, \ldots, q_e\}$ to f': $f'(V_j, M_j^\star)$, it always returns a random block unless that block has appeared for any $f'(V_j, \cdot)$ call. Since game 1 would have terminated if a collision on V_j has occurred, we can assume that if $V_j = V_i$, then $(N_j, A_j, M'_j) = (N_i, A_i, M'_i)$. The number of candidates for such collision is bounded by $(\mu - 1)$. Thus,

the probability of this collision is at most $(\mu-1)/2^n$. We take the union bound over q_e queries, we get[3]

$$|\Pr[E_3] - \Pr[E_4]| \leq \frac{(\mu-1)q_e}{2^n}.$$

Game 3: $\mathcal{B}$ terminates the game if any pair of queries generate the same tag T. Since $\mathcal{A}$ does not repeat queries and the game is not terminated due to a hash collision, the inputs to the random function must be unique. Thus,

$$|\Pr[E_2] - \Pr[E_3]| \leq \frac{\binom{q_e}{2}}{2^n}.$$

Since T is generated using a random function with unique inputs, it is indistinguishable from random blocks. If T never repeats, then C' is also indistinguishable from random blocks. Thus, $\Pr[E_3] = 0$. The overall, bound is given by

$$\varepsilon_2 + \varepsilon_3 + \Pr[E_3] + \sum_{i=0}^{2} |\Pr[E_i] - \Pr[E_{i+1}]| \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \varepsilon_4 + \frac{(\mu-1)q_e}{2^n} + \frac{\binom{q_e}{2}}{2^n}.$$

□

If the scheme is deterministic (N is set to a constant), then the bound becomes

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \frac{(2^{m+1}-2)q_e}{2^n} + \frac{\binom{q_e}{2}}{2^n} \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \frac{2q_e}{2^{n-m}} + \frac{\binom{q_e}{2}}{2^n}.$$

and if m is relatively large, $2q_e/2^{n-m}$ could be significantly higher than $\binom{q_e}{2}/2^n$. This is because any triplet (N, A, M) can be repeated at most $2^{m+1}-1$ times; the number of possible values of $M^\star$. Thus, even though the privacy bound is only up to the birthday bound in the output size of the PRI, it is still useful to use a nonce. Besides, the output of the PRI is not just the ciphertext expansion.

7 Conclusion

In this paper, we study the applications of PRIs in building flexible cryptographic modes. We show how they can be combined with leakage-resilient value comparison to build leakage-resilient MACs whose security can be adjusted based on the required level of security and implementation overhead. We also show how to use them to build succinctly committing AEAD from scratch in both the online AE and MRAE settings. scoAE particularly is an appealing construction as the specification of many existing AEAD scheme, including Ascon [DEMS21] can be adjusted to match scoAE and become succinctly committing. We believe such modification of Ascon is interesting and can be a potential future work.

Acknowledgment

The author is funded by the Wallenberg-NTU Presidential Postdoctoral Fellowship. I would like to also thank Francesco Berti and Chun Guo for their comments on the issue in LRMAC1's interpretation.

[3]The restricted PRP-PRF-switching result that is used in the proofs of Theorems 4 and 8 was first introduced in [IKMP20], to the best of our knowledge.

References

[ADG+22] Ange Albertini, Thai Duong, Shay Gueron, Stefan Kölbl, Atul Luykx, and Sophie Schmieg. How to Abuse and Fix Authenticated Encryption Without Key Commitment. In Kevin R. B. Butler and Kurt Thomas, editors, *USENIX Security Symposium*, pages 3291–3308. USENIX Association, 2022.

[BBC+20] Davide Bellizia, Olivier Bronchain, Gaëtan Cassiers, Vincent Grosso, Chun Guo, Charles Momin, Olivier Pereira, Thomas Peters, and François-Xavier Standaert. Mode-Level vs. Implementation-Level Physical Security in Symmetric Cryptography - A Practical Guide Through the Leakage-Resistance Jungle. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO I*, volume 12170 of *Lecture Notes in Computer Science*, pages 369–400. Springer, 2020.

[BF18] Manuel Barbosa and Pooya Farshim. Indifferentiable Authenticated Encryption. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, pages 187–220, Cham, 2018. Springer International Publishing.

[BGP+20] Francesco Berti, Chun Guo, Olivier Pereira, Thomas Peters, and François-Xavier Standaert. TEDT, a Leakage-Resist AEAD Mode for High Physical Security Applications. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2020(1):256–320, 2020.

[BGPS21] Francesco Berti, Chun Guo, Thomas Peters, and François-Xavier Standaert. Efficient Leakage-Resilient MACs Without Idealized Assumptions. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT II*, volume 13091 of *Lecture Notes in Computer Science*, pages 95–123. Springer, 2021.

[BH22] Mihir Bellare and Viet Tung Hoang. Efficient Schemes for Committing Authenticated Encryption. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT II*, volume 13276 of *Lecture Notes in Computer Science*, pages 845–875. Springer, 2022.

[BH24] Mihir Bellare and Viet Tung Hoang. Succinctly-Committing Authenticated Encryption. Cryptology ePrint Archive, Paper 2024/875, 2024. https://eprint.iacr.org/2024/875.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.

[BLLS22] Jannis Bossert, Eik List, Stefan Lucks, and Sebastian Schmitz. Pholkos - efficient large-state tweakable block ciphers from the AES round function. In Steven D. Galbraith, editor, *Topics in Cryptology - CT-RSA 2022 - Cryptographers' Track at the RSA Conference 2022, Virtual Event, March 1-2, 2022, Proceedings*, volume 13161 of *Lecture Notes in Computer Science*, pages 511–536. Springer, 2022.

[BSL24] Francesco Berti, François-Xavier Standaert, and Itamar Levi. Authenticity in the presence of leakage using a forkcipher. Cryptology ePrint Archive, Paper 2024/1325, 2024.

[CFGI+23] Yu Long Chen, Antonio Flórez-Gutiérrez, Akiko Inoue, Ryoma Ito, Tetsu Iwata, Kazuhiko Minematsu, Nicky Mouha, Yusuke Naito, Ferdinand Sibleyras, and Yosuke Todo. Key committing security of aez and more. *IACR Transactions on Symmetric Cryptology*, 2023(4):452–488, 2023.

[DEJ+24] Chandranan Dhar, Jordan Ethan, Ravindra Jejurikar, Mustafa Khairallah, Eik List, and Sougata Mandal. Context-Committing Security of Leveled Leakage-Resilient AEAD. *IACR Transactions on Symmetric Cryptology*, 2024(2):348–370, Jun. 2024.

[DEMS21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2: Lightweight authenticated encryption and hashing. *J. Cryptol.*, 34(3):33, 2021.

[DM21] Christoph Dobraunig and Bart Mennink. Leakage-Resilient Value Comparison with Application to Message Authentication. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 377–407. Springer, 2021.

[FLPQ13] Pooya Farshim, Benoît Libert, Kenneth G. Paterson, and Elizabeth A. Quaglia. Robust Encryption, Revisited. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *PKC*, volume 7778 of *Lecture Notes in Computer Science*, pages 352–368. Springer, 2013.

[GIK+21] Chun Guo, Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. Romulus v1. 3. *Submission to NIST Lightweight Cryptography*, 2021.

[GLR17] Paul Grubbs, Jiahui Lu, and Thomas Ristenpart. Message Franking via Committing Authenticated Encryption. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO III*, volume 10403 of *Lecture Notes in Computer Science*, pages 66–97. Springer, 2017.

[HKR15] Viet Tung Hoang, Ted Krovetz, and Phillip Rogaway. Robust authenticated-encryption aez and the problem that it solves. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015*, pages 15–44, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[IKMP20] Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. Duel of the titans: The romulus and remus families of lightweight AEAD algorithms. *IACR Trans. Symmetric Cryptol.*, 2020(1):43–120, 2020.

[JNPS21] Jérémy Jean, Ivica Nikolic, Thomas Peyrin, and Yannick Seurin. The deoxys AEAD family. *J. Cryptol.*, 34(3):31, 2021.

[Kha24] Mustafa Khairallah. CCA Security with Short AEAD Tags. *IACR Communications in Cryptology*, 1(1), 2024.

[KSW23] Juliane Krämer, Patrick Struck, and Maximiliane Weishäupl. Committing AE from Sponges: Security Analysis of the NIST LWC Finalists. Cryptology ePrint Archive, Paper 2023/1525, 2023. https://eprint.iacr.org/2023/1525.

[NSA+23] Motoki Nakahashi, Rentaro Shiba, Ravi Anand, Mostafizar Rahman, Kosei Sakamoto, Fukang Liu, and Takanori Isobe. Ghidle: Efficient large-state block ciphers for post-quantum security. In Leonie Simpson and Mir Ali Rezazadeh Baee, editors, *Information Security and Privacy - 28th Australasian Conference, ACISP 2023, Brisbane, QLD, Australia, July 5-7, 2023, Proceedings*, volume 13915 of *Lecture Notes in Computer Science*, pages 403–430. Springer, 2023.

[NSS24] Yusuke Naito, Yu Sasaki, and Takeshi Sugawara. Committing Wide Encryption Mode with Minimum Ciphertext Expansion. Cryptology ePrint Archive, Paper 2024/1257, 2024. https://eprint.iacr.org/2024/1257.

[RS06] Phillip Rogaway and Thomas Shrimpton. A Provable-Security Treatment of the Key-Wrap Problem. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 373–390. Springer, 2006.

[SPS+22] Yaobin Shen, Thomas Peters, François-Xavier Standaert, Gaëtan Cassiers, and Corentin Verhamme. Triplex: an Efficient and One-Pass Leakage-Resistant Mode of Operation. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2022(4):135–162, 2022.

[SW24] Patrick Struck and Maximiliane Weishäupl. Constructing Committing and Leakage-Resilient Authenticated Encryption. *IACR Trans. Symmetric Cryptol.*, 2024(1):497–528, 2024.